PhD Thesis

Topology Optimisation in Crashworthiness Design via Hybrid Cellular Automata for Thin Walled Structures.

Stephan Hunkeler
School of Engineering and Materials Science
Queen Mary University of London

September 2013

Abstract

Crashworthiness design is one of the most critical areas of automotive design. It is really demanding for the structure and can therefore have a large influence on the final design. It is also difficult to model accurately and costly to simulate which has an important impact on the design process. Most car companies have now stopped addressing crashworthiness design with trial and error approaches, in favour of more advanced automated structural optimisation methods. While most relevant applications so far use size or shape optimisation, the ultimate way to achieve significant mass reduction is to use topology optimisation. However, topology optimisation methods for crashworthiness design are still a work in progress. Due to the high non-linearity of crash simulations, well-established classic topology optimisation methods cannot be applied directly to crashworthiness design. Alternative methods have been and keep being developed such as the Equivalent Static Loads method, the Ground Structure Approach or the Hybrid Cellular Automata (HCA).

This thesis introduces an adapted version of Hybrid Cellular Automata using thin-walled ground structures. It combines the advantages of computing a real crash simulation while producing as an output a thin walled based topology needing minimal post-processing effort to be translated into a realistic design. In this method, the topology optimisation domain is filled up with a ground structure of thin walls which constitutes the elementary cells of the HCA method. These macro-elements replace the solid mesh elements used in the classic HCA approach. The details and implementation of the method are presented and discussed. Different application examples are detailed, including defining reinforcement patterns within extruded beams. Enriched space fillings patterns are studied and industrial application examples are presented. Eventually, recommendations for further studies and applications of the method are given.

Acknowledgements

Naming all the people who influenced your thesis is not an easy task. A lot of people should be mentioned for a lot of different reasons. A PhD thesis is not only a representation of your contribution of this specific branch of that specific field of research. It is also a testimony of an important part of your life and that should be taken into account here. This does not necessarily mean I will mention everyone from my landlord to my butcher. It just means that I want to go a bit further in my acknowledgements that mentioning those who helped me put the equations and figures together.

Without further ado, I will start with Professor Fabian Duddeck who has been the instigator of all this. I am really thankful for the opportunity he gave me to pursue this PhD and for the confidence he showed in letting me explore my own ideas. Still at Queen Mary, my second supervisor Dr Jens Müller should be mentioned along with Jonathon Hills who provided all the necessary administrative support. To all the people who shared office G24 at one point or another "I am thankful for laughter, except when milk comes out of my nose". Percy, Neveen, Henry, Shenren and later Andrei, Jerome, Hassan, Yang and Wei, it has been a pleasure sharing these moments with you. Neveen, thank you for letting us your desk, it proved to be an amazing ping-pong table. I did not forget you Faidon, but for all we shared outside G24, I thought you needed extra attention. And obviously Milan, going through this PhD with remote supervision has been a challenge and I am not sure it would have been possible without us helping each other out, so I am greatly thankful for that.

On a really practical note, I would like to thank both the School of Engineering and Materials Science and the company SFE GmbH for funding my research. I am really glad for the support and input from SFE GmbH in this project. I should mention particularly Hans Zimmer, the CEO of the company and Michel Schäfer for his enthusiasm and helpful feedback. During conferences or other meetings here and there I have had some fruitful exchanges about my research with different people. Among them, I want to mention Christophe Bastien and Jesper Christensen from Coventry University, Dr Heiner Müllerschön from Dynamore GmbH, Dr Willem Roux from LSTC or Nikola Aulig from Honda Research Institute GmbH. I also want to thank Pablo Lozano from TU München who kindly ran some computations for me in Munich. Not to forget Arup UK and especially Katherine Groves for the LS-DYNA licensing.

As I said before, I do not want to overlook the people who helped me get through this outside the office. My parents Patrick and Sylvie and my sister Nathalie. All the people who shared a roof with me along the process, Gideon, Gabrielle, Helen, Martin, Claire and Hayley. And last but not least, Jennifer who embarked an already fast moving train and made sure I did not just give it all up to open a tacos truck.

Table of Contents

Abstract i
Acknowledgements ii
Table of Contents iii
List of Figures vii
List of Tables xiii
List of Symbols xiv

Introduction **1**

1 Automotive Crashworthiness Design **3**
1.1 Motivation 3
1.1.1 Road safety 3
1.1.2 Legislation and consumer tests 4
1.1.3 New Challenges 8
1.2 Crash simulation method 10
1.2.1 Transient explicit simulation 11
1.2.2 Difficulties of numerical crash simulation 14
1.2.3 Alternative simulation methods 15

2 Structural Optimisation applied to Crashworthiness Design **18**
2.1 Structural optimisation 19
2.1.1 The optimisation problem 19
2.1.2 Different fields of structural optimisation 20
2.2 Classification of optimisation methods 22
2.3 Main optimisation methods for crash 23
2.3.1 Stochastic methods 24
2.3.2 Evolutionary Algorithms 24
2.3.3 Meta-heuristic methods and other nature-inspired algorithms . 25
2.3.4 Methods using meta-models 26

3 Topology Optimisation for Crashworthiness Design **28**
3.1 Equivalent Static Loads (ESL) based methods 29
3.1.1 ESL and global static loads 29
3.1.2 Nodal based Equivalent Static Loads 30

3.2 Ground Structure approach (GSA) . . . 31
3.3 Graph and heuristic based method . . . 32
3.4 Bi-directional Evolutionary Structural Optimisation (BESO) . . . 33
3.5 Hybrid Cellular Automata (HCA) . . . 34
3.6 Discussion of the available methods . . . 37

4 Hybrid Cellular Automata for Thin-walled Structures 39
4.1 Thin-walled structures for energy absorption . . . 39
4.2 Hybrid Cellular Automata for Thin-Walled Structures (HCATWS) . . . 42
4.2.1 Deriving thin-walled structures with the HCA method . . . 42
4.2.2 Space filling with thin walls . . . 43
4.2.3 Neighbourhood . . . 45
4.2.4 Global performance objective and constraints . . . 46
4.2.5 Algorithm overview . . . 47
4.3 Details of the algorithm . . . 49
4.3.1 Mass constraint . . . 49
4.3.2 Setpoint update . . . 50
4.3.3 Neighbourhood . . . 50
4.3.4 Cells update rules . . . 50
4.4 Implementation . . . 54
4.4.1 Architecture . . . 54
4.4.2 Automatic update with SFE CONCEPT . . . 54
4.4.3 User parameters . . . 56

5 Optimisation of extruded structures using HCATWS 58
5.1 Adapting the method for 2D design spaces . . . 59
5.1.1 Space filling . . . 59
5.1.2 Neighbourhood . . . 60
5.2 First example: shear-torsion static case . . . 61
5.2.1 Problem definition . . . 61
5.2.2 Optimisation with square space filling . . . 64
5.2.3 Rotation of the space filling . . . 73
5.3 Second example: axial impact . . . 74
5.3.1 Problem definition . . . 74
5.3.2 Topology optimisation results . . . 78
5.3.3 Comparison with enhanced sizing optimisation . . . 81
5.3.4 Different displacement constraint . . . 82
5.4 Third example: oblique impact . . . 85
5.4.1 Problem definition . . . 85
5.4.2 Topology optimisation results . . . 86
5.5 Fourth example: simplified pole impact . . . 89

5.5.1 Problem definition . . . 89
5.5.2 Topology optimisation with square space filling . . . 90
5.5.3 Additional shape optimisation . . . 91

6 Enriching the space filling with diagonal cells 94
6.1 Modification of the method for diagonal enriched 2D space fillings . . 94
6.1.1 Space filling . . . 94
6.1.2 Contact search . . . 95
6.1.3 Neighbourhood . . . 96
6.1.4 Algorithm . . . 97
6.2 Shear-torsion case . . . 98
6.2.1 Problem definition . . . 98
6.2.2 Optimisation results . . . 99
6.3 Axial impact . . . 101
6.3.1 Problem definition . . . 101
6.3.2 Results with uncoupled neighbourhood . . . 101
6.3.3 Results with coupled neighbourhood . . . 102
6.4 Oblique impact . . . 106
6.4.1 Problem definition . . . 106
6.4.2 Optimisation results . . . 106
6.5 Pole impact . . . 108
6.5.1 Problem definition . . . 108
6.5.2 Topology optimisation result . . . 108
6.6 Three point bending . . . 110
6.6.1 Problem definition . . . 110
6.6.2 Topology optimisation with different impact velocities . . . 110

7 Industrial applications with HCATWS 115
7.1 First application: side impact . . . 115
7.1.1 Problem definition . . . 115
7.1.2 Modelling and algorithm modifications . . . 117
7.1.3 Optimisation results . . . 119
7.2 Second application: pole impact . . . 122
7.2.1 Problem definition . . . 122
7.2.2 Topology optimisation with square space filling . . . 123
7.2.3 Topology optimisation with diagonal enriched space filling . . . 127

8 Conclusion and future work 129
8.1 Achievements . . . 129
8.1.1 Topology optimisation framework . . . 129
8.1.2 Design space and space filling . . . 129
8.1.3 Topology optimisation for static load cases . . . 130

8.1.4 Topology optimisation for crashworthiness design 130
8.2 Further research 131
8.2.1 Implementation 131
8.2.2 Applications 132
8.2.3 Post-treatment of the results 132
8.3 Publications 133

Bibliography **134**

Appendix A - Implementation files **143**
A.1 Optimisation input file `hcatws_user_call.sce` 143
A.2 Computation parameters and results to extract `compute_and_extract.sce` 145
A.3 List of cells `monitoring.sce` 147
A.4 Lists of cells `cells.list` and `delcell.del` 148
A.5 Computation output files `global_output.out` and `cells_output.out` 148

Appendix B - Contact definitions for diagonal enriched 2D space filling **150**

List of Figures

1.1 Evolution of road casualties in UK and US between 1970 and 2010, number of road casualties per 100,000 inhabitants (*Source: [64]*). . . . 4
1.2 Overview of main crash regulations in Europe and USA (*Source: [23]*). 5
1.3 Euro NCAP 40% offset frontal impact (left) and US NCAP full width frontal impact (right) (*Source: [22]*). 6
1.4 Euro NCAP pole impact (left) and side car impact (right) (*Source: [22]*). 7
1.5 Euro NCAP pedestrian impact tests (*Source: [42]*). 7
1.6 Small overlap test configuration (*Source: [62]*). 8

2.1 Illustrations of sizing optimisation (*top*), shape optimisation (*middle*) and topology optimisation (*bottom*) (*Source: [16]*). 20
2.2 Example of sizing optimisation parameters for an automotive roof crush application (*Source: [95]*). 21
2.3 Example of shape optimisation parameterisations for an energy absorption beam's cross-section (*Source: [59]*). 22
2.4 Example of a topology optimisation domain (*left*) and optimisation result (*right*) for a simple static 2D problem (*figures realised using the topology optimisation application TopOpt [1]*). 22
2.5 Mutation (*left*) versus re-combination (*right*) mechanisms (*Source: [38]*). 25
2.6 Example of Response Surface 26

3.1 Topology optimisation domain (*left*) and result (*right*) for a full vehicle structure using a global equivalent static load (*Source: [25]*). . . 30
3.2 Ground Structure approach application, load case and design space (*top*), ground structure (*bottom left*) and optimised structure (*bottom right*) (*Source: [99]*). 31
3.3 Graph representation of a longitudinal structural component (*Source: [94]*). 32
3.4 Application example of the BESO method. Design space and load case (*top*) and best topology derived (*bottom*) (*Source: [58]*). 34

3.5 Hybrid Cellular Automata process for crashworthiness design (*Source: [97]*). 35
3.6 Application example of HCA method for a 2D design space (*Source: [87]*). 36
3.7 Application example of Hybrid Cellular Automata for crashworthiness design of a longitudinal component under pole impact. Material density distribution for best design (*Source: [39]*). 36

4.1 Concertina (*left*) and diamond (*right*) folding modes for cylindrical tubes after axial impact (*Source: [86]*). 40
4.2 Thin-walled tubes folding modes after axial impact (*Source: [3]*). . . 40
4.3 Undeformed (*left*) and deformed (*right*) conic shells exhibiting an inverted mode of collapse (*Source: [54]*). 41
4.4 Composite panel (*left*) reinforced with thin stiffeners (*left and right*) (*Source: [121]*). 41
4.5 Typical helicopter subfloor structure for energy absorption (*Source: [17]*). 41
4.6 Thin-walled structure derived from a topology optimisation result (see Figure 3.1) and used to perform successively topometry optimisation (*left*) and then size optimisation (*right*) (*Source: [25]*). 42
4.7 Regular and quasiregular space fillings. 44
4.8 Illustration example of cubic space filling, finite element model. . . . 44
4.9 Connections between thin walls, shared nodes along connection edge (*left*) and flanges with spot welds (*centre and right*). 45
4.10 Classic neighbourhoods for 3D solid elements (*Source: [18]*). 46
4.11 Proposed neighbourhoods for thin macro elements. 46
4.12 Algorithm overview. 48
4.13 Naming convention of cells within a box. Cells $XYZ00$, $XYZ03$, $XYZ04$ (*left*) and cells $XYZ01$, $XYZ02$, $XYZ03$, $XYZ05$ (*right*). . 55

5.1 Extruded structures: basic cross-section types (*left*) and example of a pedal cut from an extrusion (*right*) (*Source: [43]*). 58
5.2 Example of reinforcement extruded structure: metallic roof beam reinforced with extruded plastic ribs (*Source: [65]*). 59
5.3 Regular 2D space fillings: triangular tiling (*left*), square tiling (*centre*) and hexagonal tiling (*right*). 60
5.4 Alternative 2D space fillings: triangular-hexagonal tiling (*left*) and rhombille tiling (*right*). 60
5.5 Classic 2D neighbourhoods (*inspired from [18]*). 60
5.6 Neighbourhoods for extruded thin-walled structures. 61
5.7 Shear-torsion problem definition. 62

5.8 Deformation directions induced by the load case, cross-section view (*left*) and side view (*right*). 62
5.9 Deformation profile of the empty cross-section (node displacements in mm). 63
5.10 Main stress profiles of the empty cross-section (stresses in MPa). . . . 63
5.11 Best design for first optimisation run, thickness distribution (mm). . . 64
5.12 Optimisation history, first run, mass value. 65
5.13 Best design for second optimisation run, thickness distribution (mm). 65
5.14 Optimisation history, second run, mass value. 66
5.15 Deformation profile of the first run's best design (node displacements in mm). 67
5.16 Deformation profile of the second run's best design (node displacements in mm). 67
5.17 Best design for third optimisation run, thickness distribution (mm). . 68
5.18 Optimisation history, third run, mass value. 68
5.19 Best design for fourth optimisation run, thickness distribution (mm). 69
5.20 Optimisation history, fourth run, mass value. 69
5.21 Optimisation history, best run, mass value and topology evolution. . . 70
5.22 Space filling with smaller cells. 71
5.23 Best design with smaller cells, thickness distribution (mm). 71
5.24 Best design with smaller cells and lower starting thickness, thickness distribution (mm). 71
5.25 Alternative cells distribution inspired by theory of torsion, thickness distribution (mm). 72
5.26 Best design with lower displacement constraint, thickness distribution (mm). 73
5.27 Rotated square space filling. 73
5.28 Best design with rotated space filling (no reintroduction), thickness distribution (mm). 74
5.29 Best design with rotated space filling (with reintroduction), thickness distribution (mm). 74
5.30 Axial impact, problem definition. 75
5.31 Empty cross-section, final deformation, von Mises stress (MPa). . . . 76
5.32 Empty cross-section, folding initiation, von Mises stress (MPa). . . . 76
5.33 Cross shaped reinforcement, final deformation, von Mises stress (MPa). 77
5.34 Cross shaped reinforcement, folding initiation, von Mises stress (MPa). 77
5.35 Optimum design topology (cross-section), thickness distribution (mm). 78
5.36 Axial impact study of multi-cell sections with reinforced corners, comparison with square cross-section (*Source: [70]*). 79
5.37 Optimum design, final deformation, von Mises stress (MPa). 79

5.38 Optimum design, folding initiation, von Mises stress (MPa). 80
5.39 Force-displacement curves of reference cross-sections and best design. 80
5.40 Energy absorption curves for the reference cross-sections and the best design. 81
5.41 Enhanced sizing optimisation, best design, thickness distribution (mm). 82
5.42 Displacement (*top*) and mass (*bottom*) history from successive response surfaces (black lines) and validation points (red dots). 83
5.43 Best design for enhanced sizing optimisation, final deformation, von Mises stress (MPa). 83
5.44 Best design for lower displacement constraint, thickness distribution (mm). 84
5.45 Best design for lower displacement constraint, final deformation, von Mises stress (MPa). 84
5.46 Oblique impact, load case. 86
5.47 Oblique impact, empty cross-section, deformation in x-direction (mm). 86
5.48 Oblique impact, empty cross-section, deformation in y-direction (mm). 86
5.49 Oblique impact, cross reinforcement, deformation in x-direction (mm). 87
5.50 Oblique impact, cross reinforcement, deformation in y-direction (mm). 87
5.51 Oblique impact, best design, thickness distribution (mm). 87
5.52 Oblique impact, best design, deformation in x-direction (mm). 88
5.53 Oblique impact, best design, deformation in y-direction (mm). 88
5.54 Oblique impact, force-displacement curves of reference cross-sections and best design (force and displacement given in impact direction). . 88
5.55 Pole impact, problem definition. 89
5.56 Pole impact, best topology, thickness distribution (mm). 90
5.57 Pole impact, best topology, maximum deformation. 90
5.58 Pole impact, additional shape optimisation, parameters and result. . . 92
5.59 Pole impact, optimisation history for force objective (RWFORC2), mass (MASS) and displacement constraint (NODOUT1). 93

6.1 Diagonal enriched space filling. 95
6.2 Uncoupled neighbourhood with diagonal cells. 96
6.3 Coupled neighbourhood with diagonal cells. 97
6.4 Diagonal enriched space filling for shear-torsion case. 98
6.5 Shear-torsion case, best design with diagonal enriched space filling, thickness distribution (mm). 99
6.6 Shear-torsion case, optimisation history of diagonal enriched space filling. 100
6.7 Shear-torsion case, evolution of active cells for diagonal enriched space filling. 100
6.8 Diagonal enriched space filling of the cross-section for axial impact. . 101

6.9 Axial impact, best design for uncoupled neighbourhood (iteration 31), thickness distribution (mm). 102
6.10 Axial impact, best design for coupled neighbourhood (iteration 81), thickness distribution (mm). 103
6.11 Optimum design topology third run, walls thickness distribution (mm).103
6.12 Axial impact, active cells for square space filling 2nd run. 104
6.13 Axial impact, active cells for coupled diagonal space filling, 1st run. . 104
6.14 Axial impact, active cells for coupled diagonal space filling, 2nd run. . 104
6.15 Axial impact, mass history for square space filling 2nd run. 105
6.16 Axial impact, mass history for coupled diagonal space filling, 1st run. 105
6.17 Axial impact, mass history for coupled diagonal space filling, 2nd run. 105
6.18 Optimum design topology for oblique impact, thickness distribution (mm). 106
6.19 Oblique impact, mass history for coupled diagonal space filling, 3rd run.107
6.20 Oblique impact, best design, deformation in x-direction (mm). 107
6.21 Oblique impact, best design, deformation in y-direction (mm). 107
6.22 Pole impact, diagonal enriched space filling. 108
6.23 Pole impact, best topology for diagonal enriched space filling, thickness distribution (mm). 109
6.24 Three point bending, problem definition. 110
6.25 Three point bending, best design for low velocity, thickness distribution (mm). 112
6.26 Three point bending, best design for low velocity, von Mises stress under deformation (MPa). 112
6.27 Three point bending, optimisation history for low velocity. 112
6.28 Three point bending, best design for intermediate velocity, thickness distribution (mm). 113
6.29 Three point bending, best design for intermediate velocity, von Mises stress under deformation (MPa). 113
6.30 Three point bending, optimisation history for intermediate velocity. . 113
6.31 Three point bending, best design for high velocity, thickness distribution (mm). 114
6.32 Three point bending, best design for high velocity, von Mises stress under deformation (MPa). 114
6.33 Three point bending, optimisation history for high velocity. 114

7.1 Side impact, problem definition. 116
7.2 Similar side impact problem studied with Graph and heuristic based method (*Source: [94]*). 116
7.3 Connections within a thin-walled cubic box. 117
7.4 SFE CONCEPT base section and user defined surfaces. 118

7.5 Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 28$^{\text{th}}$ iteration (initial state). 120
7.6 Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 100$^{\text{th}}$ iteration (initial state). 120
7.7 Optimisation history, mass value. 121
7.8 Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 80$^{\text{th}}$ iteration (undeformed state). 122
7.9 Optimisation history, mass value. 122
7.10 Full vehicle pole impact, square space fillings for reduced (*left*) and extended (*right*) design space. 123
7.11 Full vehicle pole impact, best topology for reduced (*left*) and extended (*right*) square space fillings, thickness distribution (mm). 124
7.12 Full vehicle pole impact, IED distribution for reduced (*left*) and extended (*right*) square space fillings (J.kg^{-1}). 125
7.13 Full vehicle pole impact, deformation of best design for reduced square space filling, thickness distribution (mm). 125
7.14 Full vehicle pole impact, deformation of best design for extended square space filling, thickness distribution (mm). 126
7.15 Full vehicle pole impact, alternative topology for extended square space filling, thickness distribution (mm). 126
7.16 Full vehicle pole impact, diagonal enriched space filling for reduced design space. 127
7.17 Full vehicle pole impact, best topology for reduced diagonal space filling, thickness distribution (mm). 128
7.18 Full vehicle pole impact, deformation of best design for reduced diagonal space filling, thickness distribution (mm). 128

List of Tables

3.1 Hybrid Cellular Automata variables. 35

4.1 List of user parameters 56

5.1 Parameters for first run. 64
5.2 Best designs for optimisation runs 1 to 4. 69
5.3 Influence of starting thickness on best design. 70
5.4 Piecewise linear plasticity table for impacted beam, effective plastic strain $\varepsilon^p_{\text{eff}}$ and corresponding yield stress values σ_y. 75
5.5 Results summary for axial impact problem. 85
5.6 Thickness parameters for best design and initial design (mm). 91

6.1 Common parameters for three optimisation runs. 99
6.2 Common parameters for all optimisation runs. 109
6.3 Results summary three point bending problem. 111

7.1 Parameters for first optimisation runs. 119
7.2 Parameters for optimisation with square space filling. 124

List of Symbols

$\boldsymbol{\sigma}$ Cauchy stress tensor

$\boldsymbol{\sigma}^{\nabla}$ Cauchy stress rate

ΔM_l Minimum value for $\Delta M_{f,\max}^{(k)}$

$\Delta M_{f,\max}^{(k)}$ Maximum mass fraction change possible at iteration k

$\Delta M_f^{(k)}$ Mass fraction change at iteration k

Δt^n n^{th} time step of the crash simulation

Δt_{crit} Critical time step value

$\delta t_{\max}$ Maximum thickness variation allowed per iteration

$\delta t_{\text{R}}^{(j,k)}$ Thickness increment for reintroduction of deleted cells

η, τ Function parameters for $\Delta M_{f,\max}^{(k)}$

$\lambda_{\Delta Mf}^{(k)}$ Mass fraction change modulation term at iteration k

$\lambda_{\min}$ Minimum value for the mass fraction change modulation term

$\mathbf{b}$ Body forces

$\mathbf{D}$ rate of deformation

$\mathbf{q}$ Heat flux per unit area

$\mathbf{X}$ Lagrangian coordinates vector

$\mathbf{1}_A$ Indicator function of subset A

$\mathbf{a}^n$ Acceleration vector after time step n

$\mathbf{C}^n$ Damping matrix at time step n

$\mathbf{d}^n$ Displacement vector after time step n

$\mathbf{f}^{\text{ext}}(\mathbf{d}^n, t^n)$ External forces applied to the model after time step n

$\mathbf{f}^{\text{int}}(\mathbf{d}^n, t^n)$ Internal forces applied to the model after time step n

$\mathbf{f}_{\text{eq}}^n$ Equivalent Static Load at time step n

$\mathbf{K}_{\text{L}}$ Linear stiffness matrix

$\mathbf{K}_{\mathrm{NL}}^{n}$ Non-linear stiffness matrix at time step n

$\mathbf{M}$ Mass matrix of the model

$\mathbf{v}^{n+1/2}$ Velocity vector after time step $n + 1/2$

$\mathbf{x}^{\mathrm{loc}}$ Local minimum

$\mathbf{x}^{\mathrm{opt}}$ Global minimum

$\mathcal{V}$ Volume of the structure

ν Material Poisson's ratio

Ω Whole model

ρ Material density

σ_y Yield stress

$\mathrm{disc}_{\mathrm{up}}^{(j,k)}$, $\mathrm{disc}_{\mathrm{low}}^{(j,k)}$ Extreme discrepancies between the setpoint and the cells Internal Energy levels (IED) levels

$\mathrm{d}_{\mathrm{max}}$ Maximum permissible displacement

$\mathrm{d}_{\mathrm{out}}^{(k)}$ Displacement output at iteration k

$\mathrm{f}_{\mathrm{max}}$ Force constraint

$\mathrm{F}_{\mathrm{sw,max}}$ Maximum spot weld reaction force

KE_0 Initial kinetic energy

M_0 Initial mass of the design space

$\mathrm{x}_j^{\mathrm{max}}$ Upper boundary for j^{th} design variable

$\mathrm{x}_j^{\mathrm{min}}$ Lower boundary for j^{th} design variable

θ Correlation parameter between thickness increment ζ and mass fraction change

$\tilde{U}_i^{(k)}$ Weighted sum on three previous iterations of $U_i^{(k)}$

ε_1 Convergence parameter for the inner loop

ε_2 Convergence parameter for the outer loop

$\varepsilon_{\mathrm{d}}^{(k)}$ Displacement deviation at iteration k

$\varepsilon_{\mathrm{eff}}^{p}$ Effective plastic strain

ζ Increment parameter controlling the thickness changes

ζ_{min} Minimum value for parameter ζ

a, b, c, n, $\dot{\varepsilon}_0^p$ Johnson-Cook plasticity parameters

$b_{0,u}$, $b_{q,v}$ and $b_{r,w}$ IED intervals subdivision levels

c_e Current wavespeed in element e

C_p Constraint value for p$^{\text{th}}$ global output

E Material Young modulus

e_1 Energy absorbed per unit volume

e_2 Ratio of energy absorbed over ideal energy absorption

f Objective function

g_i i^{th} constraint function

$J(\mathbf{X}, t)$ Jacobian between the current state and the reference state

$k_{\max}$ Maximum iteration number

l_0, l_q, l_r Number of IED intervals subdivisions

l_e Characteristic length of element e

$m^{(j-1)}$ Mass fraction derived from the thickness updates of step $j-1$

$M_{f,0}$ Initial mass fraction

$M_{f,\min}$ Minimum mass fraction value

$M_f^{(k)}$ Mass fraction objective

n_c Number of cells in the model

n_{TS} Number of time steps in the crash simulation

n_i Number of neighbours of cell i

$O_p^{(k)}$ p$^{\text{th}}$ global output at iteration k

$S^*_{\min}$, $S^*_{\max}$ IED setpoint limits

$S_t^{\sigma D}$ Constitutive functional of the deformation history

t_0 Initial cell thickness value

t_{E} End time of crash simulation

$t_{\min}$, $t_{\max}$ Cells thickness boundaries

t_{virt} Lower boundary for thickness parameters in enhanced sizing optimisation problems.

$U_i^{(k)}$ IED of cell i at iteration k

V^{loc} Neighbourhood of $\mathbf{x}^{\text{loc}}$

W External work

w^{int} Internal energy per unit mass

$W_{\max}$ Ideal energy absorption

$X = \bigcup_{1 \leq j \leq n} \left[\mathrm{x}_j^{\min}, \mathrm{x}_j^{\max} \right]$ Design space

$\mathbf{x}$ Design variables vector

Introduction

In the middle of 2010s, automotive industry faces a few tough challenges either commercial or technical. In an ever increasing effort on lowering CO_2 emissions of their vehicles, OEMs need to develop new models with better energy-efficiency. Whether this means going for electric cars, hybrid cars or low consumption combustion engine cars, a universal way to improve energy-efficiency is to reduce the mass of vehicles. This can be achieved by optimising the structure of cars, using different materials, dedicated metal sheet thickness, optimising the shape of components and the topology of the structure. This is not an easy task, as different design criteria need to be assessed for the optimisation (dynamic behaviour, Noise Vibrations and Harshness (NVH), aerodynamics, crashworthiness...). These criteria are often conflicting and need to be arbitrated between each others.

This thesis focuses only on crashworthiness design as it is one of the most demanding design cases for the car structure. Thanks to growing consumer awareness, car companies dramatically improved the crashworthiness of the vehicles they were producing in the late 1990's and early 2000's. The cost of this improvement was an important increase in the mass of the vehicles due to the stronger and heavier energy absorption components. Since then, car companies have tried to maintain the safety levels already achieved, while reducing the mass necessary to achieve them.

Improvements in crashworthiness are related to the improvements in design methods. Until recently, crashworthiness design remained mainly an empirical activity based on crash simulation (either numerical or physical) and iterative design changes. With the latest improvements on the accuracy of numerical crash simulation and the ever increasing computational power, more advanced systematic design methods became available. Engineers can now use optimisation methods for crashworthiness design. Such methods have been developed for a while now for sizing optimisation of components and metal sheet panels and for shape optimisation more recently. Yet, the more promising topology optimisation methods are still a work in progress when it comes to crashworthiness design.

This thesis considers the already existing methods for topology optimisation in crashworthiness design. An alternative method is proposed here in order to address some of the deficiencies of the already existing methods. This method derives a topology from a pre-defined thin-walled ground structure. Therefore, the result of the process

can be used directly as an operational design.

After introducing the context and challenges of crashworthiness design in Chapter 1, structural optimisation applied to this specific domain is discussed in Chapter 2. The available methods for topology optimisation in crashworthiness design are then presented and discussed in Chapter 3. The new method proposed in this thesis is introduced and detailed in Chapter 4. The method is used to derive the topology of extruded structures for different load cases in Chapter 5. In order to improve the method, a more complex initial ground structure is developed in Chapter 6 and used to address the same problems as in Chapter 5. More complex industrial applications are then presented in Chapter 7. Eventually, the main achievements of the thesis are summarised and future developments of the method are discussed in Chapter 8.

Chapter 1

Automotive Crashworthiness Design

The optimisation method detailed in this thesis and especially in Chapter 4 is tailored to improve the automotive development process with respect to crashworthiness. To understand this approach and its usage, it is necessary to resume roughly the basic aspects of this field, which is done in the first chapter.
In Section 1.1, a rapid overview of the motivation for crashworthiness design is given. First, some aspects of road safety and accidents involving cars are discussed, then the main legal and consumer requirements related to automotive crashworthiness are briefly presented and eventually, some of the more recent challenges related to automotive crashworthiness design are considered.
In Section 1.2, the relevant features of crash simulations are presented. The most common simulation method and the related difficulties are discussed first, and then, alternative simulation methods are detailed.

1.1 Motivation

1.1.1 Road safety

Since the introduction of the first cars in the late 19^{th} century, road safety has been a serious issue. Even though road casualties decrease steadily in Western countries such as United Kingdom and USA (see Figure 1.1), the occurrence of accidents all over the world is still really important. According to a report by the World Health Organization [125], worldwide road traffic accidents account for more than 1.2 million deaths per year and 20 to 50 million non-fatal injuries. Also, with the rapid increase of automotive adoption in middle income countries, this number could grow to 1.9 million by 2020.
To improve road safety, one can work on two main levels, active and passive safety. Active safety gathers all the measures taken to avoid accidents. This can be done through legislation (speed limits, drink driving controls...), raising awareness (both for drivers and pedestrians), improving road equipments (lightings, signalisation,

quality of road surface...) and eventually improving the vehicle (handling, breaking, detection of hazards...). On the other hand, passive safety aims to mitigate the effects of a collision. It usually involves adjusting the behaviour of the vehicles' structures in order to protect both the passengers and the pedestrians in the different crash scenarios.

Accident and safety statistics can help focus the action in relevant areas. A lot of countries are still behind in terms of road legislation. For instance, more than half of all countries require specific child restraint systems, but these countries represent only 32% of the world's population [125]. Accidents data can also help to improve the roads locally in places where accidents occur more frequently. Finally, legislators use accident statistics to define crash test requirements and adapt them to the most likely crash scenarios.

The focus of this thesis is on passive safety. Specifically, the main methods used to design the structure of cars for better crashworthiness will be discussed.

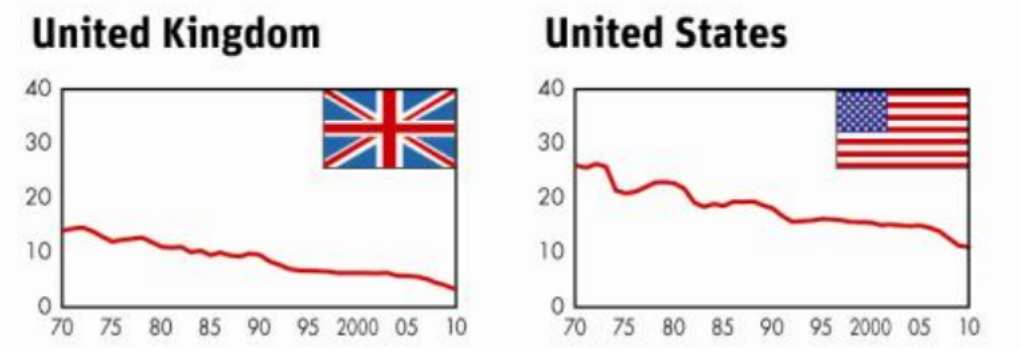


FIG. 1.1 – Evolution of road casualties in UK and US between 1970 and 2010, number of road casualties per 100,000 inhabitants (*Source: [64]*).

1.1.2 Legislation and consumer tests

To design a car for better crashworthiness, car companies usually follow a limited set of crash scenarios defined by both the regulatory and the consumer tests.

The regulatory tests are defined by each country or group of countries and need to be passed in order to sell a car in the corresponding geographical area. The consumer tests are defined by independent organisations (such as Euro NCAP[1] in Europe). Although not mandatory, they are really important for marketing purposes as their results are easily available to customers. Also, for similar crash scenarios, these tests are usually more demanding than the corresponding regulatory tests. Therefore, their influence on car design is really important.

[1] `www.euroncap.com`

Both the regulatory and consumer tests mimic likely scenarios of crash and try to cover a broad range of possibilities (see Figure 1.2). This includes for instance frontal impact (high or low speed, against a rigid wall or a deformable barrier), lateral impact, rear impact, rollover and pedestrian impact. In this last case, the injuries to the pedestrian are assessed, while in most of the other cases, the passengers' safety is the main design criterion. Also, in some specific cases such as low velocity impact, the focus is not on the passengers' safety but on the damages incurred to the car.
In the following paragraphs, a more detailed description of the three main tests categories is given.

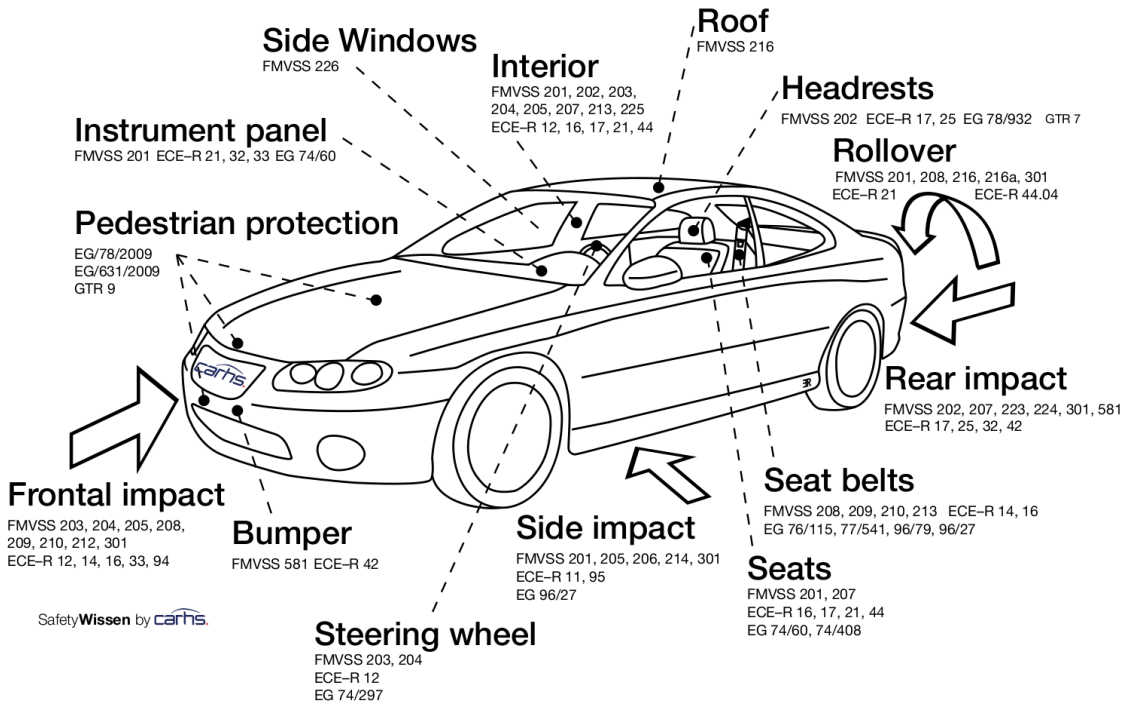


FIG. 1.2 – Overview of main crash regulations in Europe and USA (*Source: [23]*).

Frontal impact

Currently, frontal impact testing involves two main types of tests, one with a full width rigid wall and another with a 40% offset deformable barrier (ODB). Notice that the American legislation [90] actually defines different full width tests with different collision angles. For simplification, only the straight on collision is discussed here. The first main test type involves the full frontal structure of the car and leads therefore to a short crash duration with small intrusions and high decelerations for the occupants. This test is mainly designed to assess the restraint systems of the vehicle. Comparatively, the 40% offset test generates a lower deceleration and a longer crash duration. It leads to higher intrusions and is therefore more challenging for the car structure.
Figure 1.3 illustrates these two frontal tests as defined by Euro NCAP (40% offset) and US NCAP (full width).

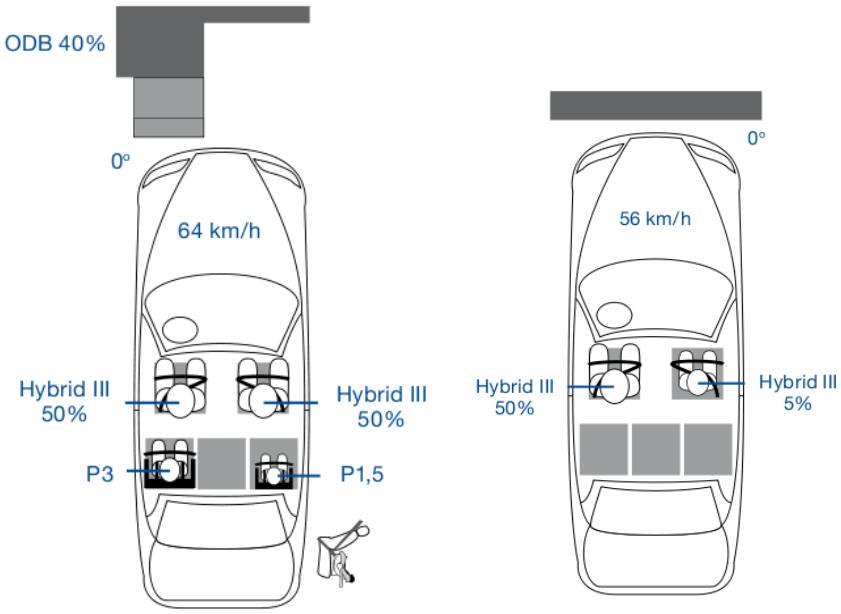


FIG. 1.3 – Euro NCAP 40% offset frontal impact (left) and US NCAP full width frontal impact (right) (*Source: [22]*).

In both cases, the test tries to assess both the damage to the structure and more importantly to the passengers. Given the difficulty to evaluate precise injuries that may be caused to the passengers, European legislation uses nine different criteria for simulated injuries on dummies in a test situation (see [44]). These are the Head Injury Criterion, the Neck Injury Criterion, the Neck Bending Moment, the Thorax Compression Criterion, the Thorax Viscous Criterion, the Femur Force Criterion, the Knee Translation, the Tibia Index and the Tibia Compression Force Criterion. These criteria assess the severity of the crash on each part of the body depending on accelerations, moments, deformations and forces applied to the dummies during the event of the crash. Different dummies are used to replicate different morphologies. Hybrid III 50% represents an average male adult being defined as the 50^{th} percentile of the American population. Hybrid III 5% represents the 5^{th} percentile. P3 and P1.5 represent respectively a 3 year old child and a 1.5 year old infant.

Lateral impact

Lateral impact is the second most important type of crash tests. For example, Euro NCAP includes two lateral impact tests (see Figure 1.4). One where the vehicle impacts a rigid pole at a speed of 29 km/h and a second one where the vehicle is impacted by a Moving Deformable Barrier (MDB) at a speed of 50 km/h. The first case simulates the event of a car leaving the road and impacting a tree or a light post. The second one simulates the event of a second car impacting the tested vehicle from the side.

With a shorter space to absorb the energy of the impact, these tests involve the passengers (here represented by adult ES-2 and children P1.5 and P3 dummies)

more directly. However, with less kinetic energy to absorb than the frontal tests, they are still manageable. Most regulatory or consumer test organisations include one or both variations of these two tests.

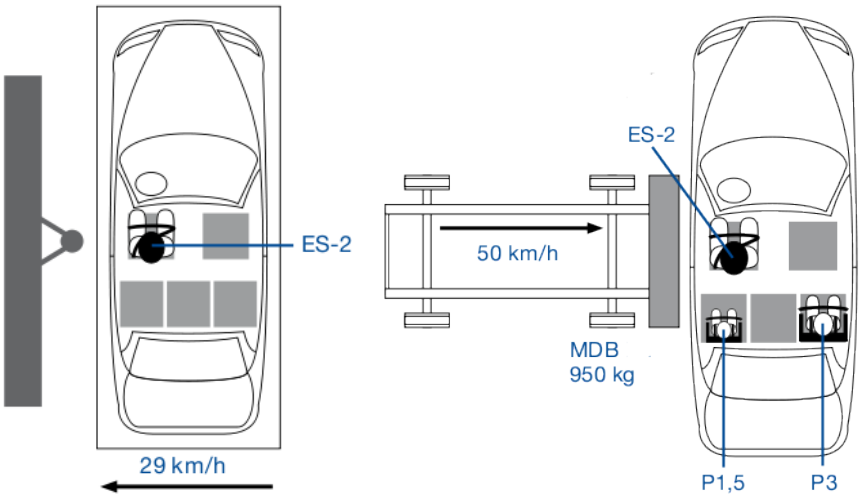


FIG. 1.4 – Euro NCAP pole impact (left) and side car impact (right) (*Source: [22]*).

Pedestrian tests

Unlike the two previous impact cases, the pedestrian tests assess the injuries incurred to a pedestrian when impacted by the vehicle. Euro NCAP defines a series of tests simulating a frontal impact between a car with a speed of 40 km/h and a pedestrian (see Figure 1.5). They assess both the possible leg injuries caused by the car's bumper or bonnet and the head trauma induced by the car's bonnet or windscreen. As the kinematics of pedestrian impacts are really complicated and quite hard to predict (see [24, 32]), these tests are performed on body components (leg and head) instead of full dummies. Different impact regions are tested in order to cover most possible scenarios.

FIG. 1.5 – Euro NCAP pedestrian impact tests (*Source: [42]*).

1.1.3 New Challenges

Automotive crashworthiness design is a really dynamic design field. The following paragraphs discuss some of the newest and future challenges. They show that it is essential to provide appropriate design methods, including appropriate optimisation methods for car components or for the full car.

New tests and ratings

In an effort to keep increasing the level of safety, the consumer and regulatory tests are permanently re-discussed and updated. These discussions are led by different organisations such as the European Enhanced Vehicle-safety Committee (EEVC)[2]. Whether it is small modifications to already existing tests, changes in the tests ratings to put more emphasis on a given test or even implementing a whole new test, this is an ongoing challenge for every car company.
The small overlap test introduced by the Insurance Institute for Highway Safety (IIHS) in 2012 is an example. This is a high speed frontal impact similar to the ones described in the previous section. In this case, the vehicle travels at 64 km/h and impacts a rigid barrier with a 25% offset (see Figure 1.6). According to the first results published by IIHS [63], this new test is really demanding for the car structure and a lot of cars highly rated in other usual tests performed badly in this test. With a smaller overlap than the 40% of the classic overlap test, the main energy absorption parts of the car structure may or may not be activated which completely changes the kinematics and therefore the outcome of the impact.

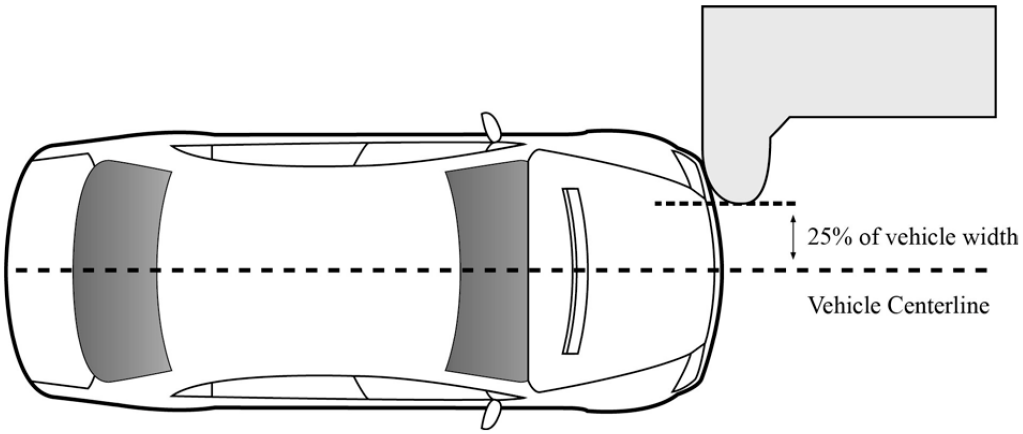


FIG. 1.6 – Small overlap test configuration (*Source: [62]*).

Another example of the evolution of tests can be found in the Euro NCAP ratings. With most vehicles tested getting good results for adult occupant protection, Euro NCAP is slowly putting more emphasis on child occupant and pedestrian protection (see [41]). As a result, car companies, need to adapt and follow this trend for the design of their future models.

[2] `www.eevc.org`

New structural concepts

The evolution of car concepts is also a challenge with regards to crashworthiness design. Most combustion engine cars use a similar architecture with a front engine. This has strongly dictated car design for frontal impact. With newer concepts such as electric cars, the engines can be smaller and be positioned in unusual places (for example in the back, in the wheels). This is an opportunity to design the cars' front ends in a completely different way.

Nevertheless, electric cars and hybrid cars introduce a new problematic. They include batteries which, in some cases, can be really large. For safety reasons, those batteries need to be protected from impact which is clearly a design constraint. As a result, Euro NCAP, for instance, has defined specific measures to test and assess electric vehicles (see [40]). The idea is to make sure that these vehicles do not present any risk of electrical shock after a crash, both for the passengers and for the emergency services.

Ultimately, car companies are trying to expand their platform strategy. In order to reduce manufacturing cost, they develop a single vehicle platform which is meant to be used by a number of different vehicles. These vehicles vary in length, height, weight and other characteristics. Still, they are based on a single platform, which should guarantee the best crashworthiness for various vehicles. As an illustration, Volkswagen Group recently developed its platform MQB[3] for supermini cars (Volkswagen Polo) to large family cars (Volkswagen Passat) of the main brands of the group (Volkswagen, Audi, Seat and Skoda). Similarly, PSA Peugeot-Citroën unveiled its new mid-size platform EMP2[4] for small family cars (Peugeot 308) to small MPVs[5] (Citroën C4 Picasso) of both PSA Peugeot-Citroën and General Motors brands.

New lightweighting strategies

In order to meet fuel consumption targets, car companies need to use every option available. Reducing the mass of vehicles is one option, but it is directly conflicting with crashworthiness performance. Energy absorbing structures are heavy and companies need to find appropriate methods to reduce the mass of their vehicles while ensuring good crashworthiness performance.

Use of lighter materials such as Carbon Fibre Reinforced Plastics (CFRP) is becoming increasingly popular. Nevertheless, the numerical modelling of this material under high deformation is still a challenge. Other lightweighting strategies are also currently used. They can involve a modification of the manufacturing and joining methods or optimisation of the structure. The use of optimisation tools to design more efficient components or even to optimise the whole car assembly is key to

[3] From German *Modularer Querbaukasten*, which can be translated by *Modular Transversal Toolkit*

[4] *Efficient Modular Platform 2*

[5] *Multi-Purpose Vehicle*

achieve significant mass reduction. Specific optimisation methods need to be used for crashworthiness design (see Chapter 2).
Also, with the development of new vehicle concepts such as urban two-seater cars (e.g. Renault Twizy), more vehicles are bound to fall in the category of heavy quadricycles (category L7E, as defined in [45]). These vehicles do not belong to the same category as usual cars and therefore they do not legally have to satisfy any crash tests. Still, with the development of this vehicle category, it is important that the legislators and consumer tests programmes develop new tests, adapted to that specific category of vehicles. This would ensure a minimum safety level for these vehicles.

Shortened development cycle

A final challenge is dictated by the competition between car companies. In order to stay ahead of their competitors, car companies have drastically shortened the development cycle of their new models. As a result, they need quicker and more efficient design methods, especially when it comes to crashworthiness design.
Different design methods can be used in the different phases of the vehicle development. In the concept phase or early design phase, exploratory design methods are needed. In the product development phase, targeted design methods need to be used in order to design either at the component level as for the full car. Later in the product development, design methods need to enable a fine tuning of the vehicle specifications in order to validate the design before mass production.

1.2 Crash simulation method

As in most industrial applications, the complexity of the analysis and the prohibitive cost of physical experiments have led automotive engineers to use numerical simulation in their design process. This is true, amongst other things, for the simulation of crash. It allows multiplying the designs tested and enables cars manufacturers to build an optimisation strategy that would not be affordable otherwise. Ideally, the physical tests should only validate the numerical simulations.
With its ability to deal with large deformations, high strain rates, transient problems and contact, the explicit finite element method is ideally adapted for crash simulation. The method is outlined in 1.2.1 and some of the related difficulties are discussed in 1.2.2. Other simulation methods are also available. Some of the simplified simulation methods used in early-phase design are introduced in 1.2.3.

1.2.1 Transient explicit simulation

Finite element discretisation and governing equations

Crashworthiness design involves large deformations of solid structures. Therefore, it is appropriate to adopt non-linear transient methods for numerical simulation. For more details on these methods and the different formulations (Eulerian, total Lagrangian and updated Lagrangian for instance), the reader can refer to [13] and particularly Chapter 4 for Lagrangian meshes.

The structure is discretised using finite elements. Classic 3D solid elements (or brick elements) are the most natural way to discretise a 3D model. Yet other types of elements are available and can sometimes prove really useful to simplify a model and therefore save computation time. 2D shell elements are commonly used to model thin-walled structures. 1D beam elements can be used to model connections. In most finite element models for crash simulation, a mix of the different element types is used.

Whichever element type is chosen, the mechanical behaviour of the model must satisfy the following governing equations (here expressed for the Updated Lagrangian formulation).

Conservation of mass:

$$\rho\left(\mathbf{X},t\right)J\left(\mathbf{X},t\right)=\rho_0\left(\mathbf{X}\right)J_0\left(\mathbf{X}\right)=\rho_0\left(\mathbf{X}\right) \tag{1.1}$$

Where $\mathbf{X}$ is the Lagrangian coordinates vector, t is the time discretisation, ρ is the material density, $J\left(\mathbf{X},t\right)$ is the Jacobian between the current state and the reference state and 0 indicates the reference state.

Conservation of linear momentum:

$$\nabla\cdot\boldsymbol{\sigma}+\rho\mathbf{b}=\rho\dot{\mathbf{v}}\equiv\rho\frac{D\mathbf{v}}{Dt} \tag{1.2}$$

Where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\mathbf{b}$ the body forces and $\mathbf{v}$ the velocity.

Conservation of angular momentum:

$$\boldsymbol{\sigma}=\boldsymbol{\sigma}^T \tag{1.3}$$

Where T indicates the transposition operator.

Conservation of energy:

$$\rho\dot{w}^{int}=\mathbf{D}:\boldsymbol{\sigma}-\nabla\cdot\mathbf{q}+\rho s \tag{1.4}$$

Where w^{int} is the internal energy per unit mass, $\mathbf{D}$ is the rate of deformation, $\mathbf{q}$ is the heat flux per unit area and ρs is the heat source per unit volume.

Constitutive equation:

$$\boldsymbol{\sigma}^{\nabla} = S_t^{\sigma D}\left(\mathbf{D}, \boldsymbol{\sigma}, \text{etc.}\right) \tag{1.5}$$

Where $\boldsymbol{\sigma}^{\nabla}$ is a Cauchy stress rate and $S_t^{\sigma D}$ a functional of the deformation history.

Material modelling

Material modelling is critical in crash simulation. As the structures undergo large deformation, the perfectly elastic behaviour approximation is not valid. The constitutive models must include both elastic and plastic behaviour. Also, with the transient nature of crash, damage and fracture should be considered. Finally, the dynamic nature of crash calls for strain-rate dependent constitutive models. For a theoretical overview of the constitutive models used in crash simulation the reader can refer to Chapter 5 of [13]. A more exhaustive list of material modelling possibilities can be found in the commercial software manuals (e.g. [78, 81] for LS-DYNA). With the different materials used within the car's structure, different constitutive models with different complexity level can be used in the same finite element model. It should also be noted that in the context of topology optimisation, treated from Chapter 3 of this thesis, material modelling is especially important. A lot of topology optimisation methods rely on density methods with interpolations of the material properties. In this case, elements with low density are assigned non-physical material properties which can lead to unstable simulations.

Explicit integration scheme

In order to solve the transient problem, the mechanical problem is divided into time steps. Going from a time step to the next one is done through an integration scheme. The most common integration scheme for explicit methods is the central difference method.

Let the end time of simulation be t_{E}, the simulation is divided into n_{TS} time steps Δt^n (for $1 \leq n \leq n_{\mathrm{TS}}$). The time at time step n is t^n and the displacement is $\mathbf{d}^n$ (displacement vector through the nodes of the model). The time increment to go from time n to $n+1$ is defined as $\Delta t^{n+1/2} = t^{n+1} - t^n$.

Using the central difference method, the velocity $\mathbf{v}^{n+1/2}$ can be expressed as:

$$\mathbf{v}^{n+1/2} = \mathbf{v}^{n-1/2} + \Delta t^n \mathbf{a}^n. \tag{1.6}$$

Where $\mathbf{a}^n$ is the acceleration vector after time step n. Combined with the equation of motion at step n:

$$\mathbf{M}\mathbf{a}^n = \mathbf{f}^n = \mathbf{f}^{\text{ext}}\left(\mathbf{d}^n, t^n\right) - \mathbf{f}^{\text{int}}\left(\mathbf{d}^n, t^n\right). \tag{1.7}$$

Where $\mathbf{M}$ is the mass matrix of the model, $\mathbf{f}^{\text{ext}}(\mathbf{d}^n, t^n)$ are the external forces applied to the model at time step n and $\mathbf{f}^{\text{int}}(\mathbf{d}^n, t^n)$ are the internal forces of the model at time step n. Notice that in Equation (1.7), the damping is included in the internal forces. The velocity update at step n is given by:

$$\mathbf{v}^{n+1/2} = \mathbf{v}^{n-1/2} + \Delta t^n \mathbf{M}^{-1} \mathbf{f}^n. \tag{1.8}$$

The actual implementation of the method is a bit more complex with successive update of the forces, accelerations, velocities and displacements, but still easy to implement. For details, the reader can refer to Chapter 6 of [13].
Nevertheless, the method is only conditionally stable as it does not solve directly the system of differential equations. To ensure stability, the time step should not exceed a critical value Δt_{crit}. This critical value is given by the Courant condition (see [30]) as:

$$\Delta t_{\text{crit}} = \min_{e \in \Omega} \frac{l_e}{c_e}. \tag{1.9}$$

Where l_e is the characteristic length of element e, c_e the current wavespeed in element e and Ω the whole finite element model. Depending on the element type, l_e and c_e have different expressions. For shell elements (plane stress), c_e is defined as

$$c_e = \sqrt{\frac{E}{\rho\,(1-\nu^2)}}. \tag{1.10}$$

where ρ is the material density, ν the Poisson's ratio and E the Young modulus. Notice that the definition of c_e given in Equation (1.10) is based on elastic properties of the material. For elastic-plastic materials, the elastic wavespeed is also used. In this case, the stability of the model should be checked after the simulation is run. An easy (but non sufficient) way to check the stability of a model is to verify that the total energy is conserved throughout the simulation.

Contact

Another important feature for crash simulation is the contact algorithm. Indeed, with the large deformations occurring during the crash, different parts of the structure come into contact with each other. Therefore, it is important to model this phenomenon.
There is a number of ways to model contact (see for instance [126]), nevertheless, for crash applications, the most popular method is the so-called "penalty method". In this method, contact is sought between finite element nodes and surfaces. When detected, the algorithm virtually places interface springs between the surface and the penetrating nodes. This virtual spring generates a force inversely proportional to the distance between the node and the surface, or penalty force. This allows to

avoid penetration of the node through the surface. The important characteristics of this method are first the contact search algorithm and secondly the penalty force definition.
Once again, contact definitions are crucial for topology optimisation. With variations of the topology during the optimisation process, new contacts can become possible between the different parts of the model. This should be considered either by the optimisation algorithm or by the initial contact definition.

1.2.2 Difficulties of numerical crash simulation

One of the characteristics of crash simulation is the size of the finite elements models used. Due to high deformation levels and contact research algorithms, the models used for crash need to have small elements to achieve a good accuracy. 5 mm size for shell elements can be given as a typical example. In this case, complete models have usually one or two million elements. With the computation time directly depending on the number of elements, this is something to take into consideration when generating the finite element mesh. Also, because the explicit integration time step depends on the smallest element in the model (see Equation (1.9)), hybrid meshes with coarser mesh in areas of low stresses have little interest in terms of reducing the computation time.
A practical method to deal with the size of these models and to reduce computation time is to use parallel computing. In this case, the model is decomposed into sub-models and a different CPU solves the problem separately for each sub-model. To ensure continuity, there needs to be communication between the boundary nodes of the different sub-models. Theoretically, this method allows to divide the computation time by the number of CPU available. However, with a high number of CPU, communication between the CPU becomes critical and the scaling of computation time becomes less favourable.
Another specificity of those models is the complex mix of different materials and joining elements in a car assembly. As a result, the finite element model contains several finite elements types and a high number of different material models. Shell elements are mainly used to model metal sheets or plastic parts of the car body. Volume elements can be used to model stiff parts such as the engine or on the opposite to model soft parts such as seat foams. They can also be used to model adhesives or spot welds. Eventually, 1D elements such as rigid or elastic beams, possibly with damping behaviour can be used to model dampers or simplified subsystems such as doors. Given the amount of different element types and material behaviours it is really easy to get confused and mix everything up in a complete model.
As just mentioned, there are always several methods to model a given part/element in the crash simulation model. Then, the difficulty is to ponder between physical accuracy and computational effort of the modelling chosen. For instance, spot welds

can be modelled with a rigid element between two nodes, which has a really low level of physical accuracy but a low computational effort. Another way to model them is to use volume elements including plastic deformation and failure criterion. That second option has a much better physical accuracy but represents a higher computational effort.
Generally, the choice of material models is a really problematic point, especially for plastic behaviour. For metals for instance, depending on the data available, one can choose between general plasticity models such as the Johnson-Cook law (see [66]), piecewise linear approximations or complex strain-rate dependent tables based on experimental data. With composite materials being increasingly popular in automotive applications, that question of material modelling gets even more crucial.
Another specificity of crash simulation is that the high strains appearing during the crash can lead to a decrease of the time step. Indeed, the deformations can change the characteristic length of the elements and therefore change the critical time step (see Equation (1.9)). Therefore, a time step control is usually needed in order to keep the time step constant and allow for faster computation time. The time step is given by the most critical element of the model and depends on its characteristic length, Young's modulus and density (see Equation (1.10)). When an element sees its characteristic length decreased, a convenient way to control the time step of the model is to artificially increase the density of this element. Yet, using this method adds mass to the whole model, therefore the user must make sure that the inertia effects of the added mass do not noticeably modify the results of the simulation.
As mentioned previously, contact is a big issue for crash simulation. In some cases, contact algorithms may account for half or more of the computation time. Once the contact algorithm is chosen, one must define the areas where contact should be checked. An easy way to do it is to check all the nodes of the model against all the surfaces. However, this is expensive and inefficient as some parts of the model will never come into contact with others. Contact groups can be defined to check contact only between relevant parts. This allows to save a lot of computation time, but it is more tedious and should be done carefully as sometimes some unexpected contacts arise.

1.2.3 Alternative simulation methods

As discussed in the previous section, crash simulation through transient explicit methods is both computationally expensive and time consuming to set-up. As a result, alternative methods to simulate crash were and are still developed. Three of the most significant ones are presented here.

Lumped masses

The lumped masses method, as for example proposed by Kamal *et al.* [67], is one of the first approaches for crashworthiness. It was developed to simulate the response of a vehicle for frontal impact. To do so, the structure is approximated by a 1D system of lumped masses and springs. Within this assumption, it is relatively easy to derive the behaviour of the structure for crash. However, it is quite difficult to define the system of lumped masses and springs in the first place. The spring parameters are normally derived from physical experiments. As a result, it is challenging to use the method to create a completely new vehicle. The position and succession of springs and masses are difficult to define and suppose a lot of experience and tuning to achieve a good correlation between experiment and simulation. Finally, when modifications are done to the lumped masses model to improve the crash behaviour, the corresponding physical modifications to do on the real model are not straightforward.
The lumped masses method has been used in different early phase design applications (e.g. [84]). Its limitations are one of the reasons for the development of the following methods (see [85]).

Plastic hinge approach and multi body modelling

The plastic hinge approach was first introduced by Nikravesh *et al.* [92] to represent the structural dynamics of road vehicles in crash situations. It is a simplified modelling of the crash (in comparison with explicit finite element analysis). The idea of the method is to focus on the local absorption of energy and global deformation. The real geometry of the vehicle is replaced by simplified structural bodies linked with kinematic joints. These joints allow to describe the kinematics of the deformation. A generalised non-linear spring element, able to represent elastic-plastic loading and elastic unloading, is used to represent the deformation and energy absorption of the structural bodies. In a sense, it can be seen as an extension of the lumped masses method.
Multi body modelling uses this method in 3D dimensions coupled with flexible bodies to account for contact issues. It is for instance implemented in the commercial software MADYMO [115]. This approach is popular for early phase design and has been applied for the optimisation of larger problems [7].

Super-Folding Element

The Super-Folding Element was developed after the work of Alexander [5]. Using experimental observations to define the kinematically admissible deformation field, he derived analytical functions for the progressive collapse of axially compressed cylindrical tubes. Based on this work, Abramowicz developed the Super Folding

Element Method (see [3]) to assess the design of thin-walled components for axial crash load cases. Using this method, the crushing characteristics of virtually any cross-section can be derived. However, the method only gives a mean crushing force value which overlooks the question of initial peak force. Also, with this mean value, it is hard to get a precise idea of the kinematics of an assembly of parts or even a full vehicle model. The method has later been extended to cover other load cases than axial loadings [114]. It is implemented in the commercial software Visual Crash Studio [60].

Using this method, Halgrin *et al.* [55] developed a hybrid approach. They couple an explicit finite element model with a Super-Folding Element model. This allows for detailed simulation (through the explicit finite element model) in some areas of interest and quicker computation (through the Super-Folding Element model) in other areas.

Chapter 2

Structural Optimisation applied to Crashworthiness Design

This thesis presents a method for topology optimisation in crashworthiness design. Before presenting the method (Chapter 4) or even talking about topology optimisation for crashworthiness (Chapter 3), it is important to define Structural Optimisation and to clarify it in the context of crashworthiness design.
Structural Optimisation is a really wide topic and quite often, a lot of optimisation methods are available to solve the same problem. It is important to have a good understanding of the design problem and its specific difficulties on the first hand and of the different optimisation methods with their capabilities and efficiency on the second hand. A given method can perform admirably for some design problems and fail to reach any satisfying solution for other problems. On that matter, an insightful discussion has been initiated by Prof. O. Sigmund in [111]. In this paper he criticised the use of non-gradient based optimisation methods in topology optimisation and proposed specific criteria for cases in which such methods should be used. This triggered discussions in one of the main scientific journal on structural optimisation and led to recommendations for new editorial guidelines to make sure only relevant methods for a given design problem would be used (see [105]).
As discussed in Chapter 1, crashworthiness simulation is really complex and computationally expensive. As a result, adapted optimisation methods should be used. In this chapter, the concept of Structural Optimisation and the terminology used in the following chapters is introduced (Section 2.1). Then a classification of optimisation methods is proposed (Section 2.2). Finally, the main optimisation methods used in crashworthiness design are presented (Section 2.3).

2.1 Structural optimisation

2.1.1 The optimisation problem

The classical constrained single objective optimisation problem can be formulated as:

$$\begin{cases} & \min \quad f(\mathbf{x}) \\ \text{for} & g_i(\mathbf{x}) \leq 0 \quad 1 \leq i \leq m \\ \text{and} & \mathrm{x}_j^{\min} \leq \mathrm{x}_j \leq \mathrm{x}_j^{\max} \quad 1 \leq j \leq n \end{cases} \tag{2.1}$$

Where f is the objective function, $\mathbf{x}$ the design variables vector, g_i are the constraint functions, $\mathrm{x}_j^{\min}$ and $\mathrm{x}_j^{\max}$ are the boundaries for the j^{th} design variable. The union X of all the intervals $\left[\mathrm{x}_j^{\min}, \mathrm{x}_j^{\max}\right]$ is called the *design space*.
Notice that maximising the function f is the same as minimising $-f$, which makes Equation (2.1) a general formulation of the optimisation problem. From here on, only the minimisation of the objective function will be discussed.
The designs $\mathbf{x}$ that effectively satisfy the set of constraints are called *feasible solutions*. A feasible solution that minimises the objective function is called *optimal solution*.
A local minimum $\mathbf{x}^{\mathrm{loc}}$ (as opposed to global minimum $\mathbf{x}^{\mathrm{opt}}$) is defined as a point for which there exists a neighbourhood V^{loc} of $\mathbf{x}^{\mathrm{loc}}$ fulfilling the condition:
$\forall \mathbf{x} \in V^{\mathrm{loc}},\ f(\mathbf{x}) \geqslant f\left(\mathbf{x}^{\mathrm{loc}}\right)$.
Generally, it is relatively easy to find a local minimum of the function. To assess that a particular solution is the global minimum, it is necessary to refer to additional knowledge of the problem (for instance the convexity of the objective function).
In [123], the authors discuss the different characteristics of objective functions and the relative difficulties to optimise each of them. A convex function represents the ideal case for optimisation as there is a guaranteed minimum and it is easy to find using gradient information. A function presenting multiple local minimum is called *multimodal*. With these functions, using only gradient information it is possible to get stuck in a local minimum. Taking into account the different constraints, the optimisation task gets even more complicated.
In practical optimisation cases, there is no global information on the objective function. The optimisation is made using only evaluations of the objective and constraint functions in different design points.
In the case of optimisation for crashworthiness of a vehicle structure, with the hundreds of parts of the structure, the optimisation task can quickly become overly complicated. In theory, multidisciplinary optimisation should be used to take into account other domains such as dynamic, Noise Vibration and Harshness (NVH), *etc.* This thesis deals only with crashworthiness design. With all the different test cases

involved (see Section 1.1.2) this is a multi load case optimisation problem.
To simplify the task, engineers can optimise components of the car separately. They can for instance choose to optimise the most influencing parts for each particular test case (*e.g.* crash boxes and front-rails for a frontal crash). But when a part is very influential for several test cases, multi load case optimisation can be performed. In this case, the problem must be solved using a multi-objective optimisation algorithm, and requires complex optimisation loops. Notice that a single load case problem can also be a multi-objective problem when different outputs need to be optimised simultaneously. The applications presented in this thesis focus on single objective optimisation test cases.

2.1.2 Different fields of structural optimisation

Structural optimisation aims at designing the best structure for a given objective and constraints. For instance, a structural optimisation problem could be formulated as "What is the best front-rail design for a given vehicle, given specific weight and reaction force constraints, to maximise the specific energy absorbed in a frontal crash test ?".
Generally, Structural Optimisation is divided into three main categories (see Figure 2.1) depending on the type of design variables involved (and therefore the type of algorithms used to solve it).

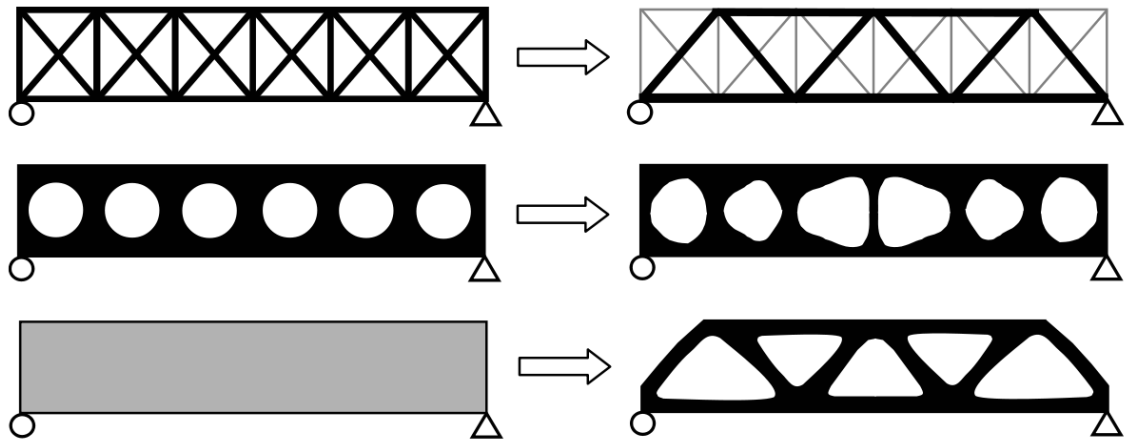


FIG. 2.1 – Illustrations of sizing optimisation (*top*), shape optimisation (*middle*) and topology optimisation (*bottom*) (*Source: [16]*).

Sizing optimisation deals only with design variables such as part's thickness (see Figure 2.2). This type of problems is easy to implement as the finite element model does not need to be modified, only the thickness values should be updated. They can usually be solved with standard optimisation algorithms (see Section 2.3).

Shape optimisation focuses on the geometry of the parts, mostly with a fixed layout (see Figure 2.3). The dependencies between the design variables and the output responses are much more complex than for sizing optimisation. Indeed, while increasing the thickness of a panel can often lead to an increase in energy absorption, it is usually more difficult to guess the effect of a shape change for the structure behaviour. Still, those problems can be solved using usual optimisation algorithms (see Section 2.3).

Topology optimisation is the field of structural optimisation which assesses how a given design space should be filled in order to optimise the objective function (see Figure 2.4).
Therefore, topology optimisation means not only to optimise the geometry of an element but also to assess its topological class (*i.e.* the geometrical layout). Compared to sizing optimisation and shape optimisation, it involves different and usually many more design variables. Alternative algorithms are needed to solve this type of optimisation problems (see Chapter 3 for crashworthiness design).

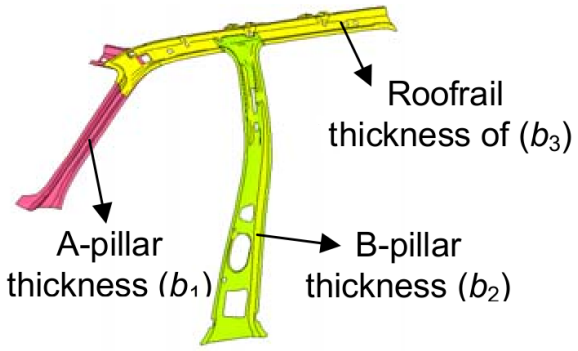


FIG. 2.2 – Example of sizing optimisation parameters for an automotive roof crush application (*Source: [95]*).

A more detailed discussion of the different types of structural optimisation is given in [76]. It focuses on the application to vehicle structure design.
In this paper, the definition of *Topology optimisation* is limited to the special case where design variables are the densities of each finite element in the model. This supposes that the finite element model contains only volumic finite elements. By varying this density from 0 to 1, the best topology of the structure can be derived.
Topometry optimisation is defined for cases where the design variables are a special property of each finite element in the model [75] (*e.g.* thickness for shell elements).
Shape optimisation is limited to the cases where the shape modifications allowed are defined directly by the user with shape parameters.
Topography optimisation is defined as a special case of *Shape optimisation* where the shape modifications are introduced by the optimisation algorithm without initial input from the user [74].

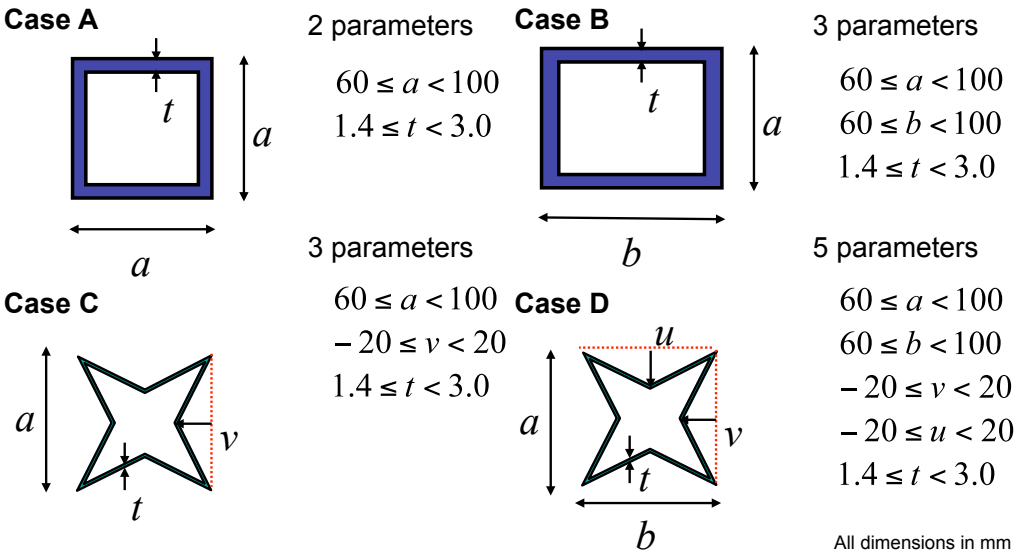


FIG. 2.3 – Example of shape optimisation parameterisations for an energy absorption beam's cross-section (*Source: [59]*).

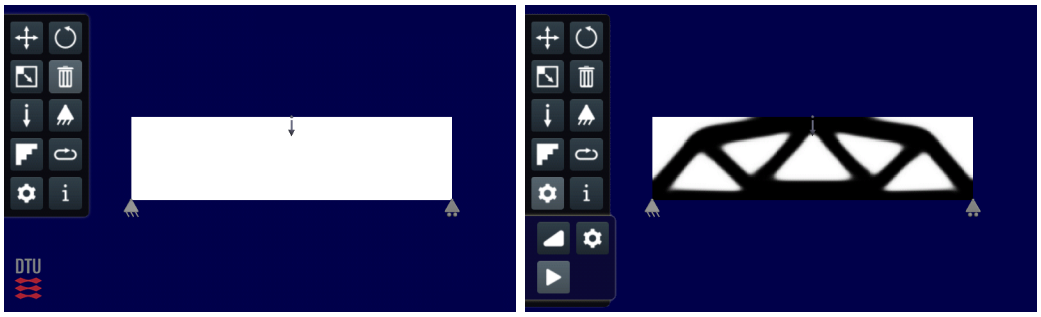


FIG. 2.4 – Example of a topology optimisation domain (*left*) and optimisation result (*right*) for a simple static 2D problem (*figures realised using the topology optimisation application TopOpt [1]*).

Freeform optimisation is defined as another special case of *Shape optimisation* where every node of the finite element model is a design variable and can move in every direction [101].
Within this thesis, the more general definition of topology optimisation (*i.e.* solving the problem of how to fill a given design space to optimise the objective function) is used.

2.2 Classification of optimisation methods

Classification of optimisation methods is not straightforward. The taxonomy used to group different methods is sometimes confusing or restrictive. In [111], the distinction is made between gradient based and non-gradient based (or derivative-free) optimisation methods. Although convenient, this classification can be judged manichean as some optimisation methods use hybrid approaches to couple advantages of each of these type of method. Also, this taxonomy fails to describe the diversity

of derivative-free optimisation methods. In [105], local and global optimisation algorithms are opposed. Global optimisation algorithms being defined as algorithms avoiding to get trapped in local optimum. Again, this classification is unsatisfying as a lot of algorithms can be used for both local and global optimisation. Other popular classifications oppose for instance individual and population-based methods or heuristic and non heuristic methods. A heuristic method being defined as a solution search method with some level of learning process. A heuristic method should provide a quick satisfying solution but does not guarantee its optimality.
Here, an alternative classification is proposed based on the type of structural information used by the optimisation method. The first category gathers optimisation methods using element-based or node-based structural information to go from one iteration to the other. This would gather any optimisation computing the sensitivities of the objective or constraint functions to the nodes' or elements' variations. This includes for instance gradient-based optimisation method for freeform optimisation (see Section 2.1.2), where every node of the finite element model is a design variable and the sensitivities or the response functions would be computed for every design variable. Similarly this could include methods for topometry optimisation or topography optimisation.
The second category does not use element-based or node-based information, only the objective and constraint functions are used to go from an iteration to the other. This would include for instance Evolutionary Algorithms (see Section 2.3.2) or Response Surface Methods (see Section 2.3.4) but also, gradient based methods if the number of design variables is limited and the gradient information of the objective and constraint functions is found through finite differences.
Hence, the main difference between these two categories is the level of understanding of the physical phenomenon which is put into the optimisation algorithm. In the first category, the algorithm has a good understanding of the influence of each element in the model on the physical phenomenon while in the second category, the physical phenomenon is more of a black box. Naturally, the first category of algorithms is more difficult to implement and usually requires more evaluations of the objective function than the second one. Therefore, for crashworthiness design applications, the second category of optimisation algorithm is more popular.

2.3 Main optimisation methods for crash

As crash simulations are computationally expensive and highly non-linear, methods using element-based structural information (see Section 2.2) are hard to apply. Even for methods based only on objective and constraint functions, it is hard to compute gradient information. This can technically be done through finite differences but the amount of crash computations needed makes it unrealistic as soon as several

design variables are considered.
Therefore, derivative-free methods based on objective and constraint functions are more popular when it comes to crashworthiness design. These methods do not generally guarantee the optimality of the solution they provide. However, they allow to improve the objective function with a limited number of function evaluations, which is already satisfying in crashworthiness design. The following sections introduce some of the most popular optimisation methods for crashworthiness design, from the simple stochastic methods to the more sophisticated meta-modelling.

2.3.1 Stochastic methods

Purely stochastic methods represent the simplest level of optimisation methods. They consist in scanning the design space with randomly generated designs in order to improve the objective function. While the Monte-Carlo method generates the designs completely randomly, the Latin Hypercube Sampling (LHS) starts by dividing the design space into equally distributed intervals (or sub-spaces) and then generates designs randomly into each sub-spaces. As a result, LHS method usually needs less samples than the Monte-Carlo method to scan the design space (see [46]).
These methods can be interesting to roughly scan a design space but remain inefficient to perform optimisation due to the high number of computations needed.

2.3.2 Evolutionary Algorithms

Evolutionary Algorithms are one of the most popular optimisation methods for crashworthiness design. They have been used for instance in [35, 56]. For a history and overview of the variety of specific algorithms, the reader can refer to [12] or [113].
Evolutionary Algorithms are inspired by evolution mechanisms of natural species. The principle is to generate a population of elements amongst which the "strongest" are going to be selected, and used to create the next generation of elements. For an unconstrained problem, a design will be judged stronger than another one if its objective value is lower than the other design (minimisation case).
The mechanisms to select designs of a given generation (or *parents*) and create new designs for the next generation (or *offsprings*) are particular to each algorithm, and may be completely different from one to another. To simplify, two main types of mechanisms can be outlined (see Figure 2.5): (1) mutation mechanism (variation around a single *parent* design), and (2) re-combination mechanism (also known as "Genetic Mechanism") which combines two (or more) different *parents* to generate an *offspring*. A combination of both mechanisms is often used to get a practical solution. In the figure, the behaviour of each mechanism type to go from a first generation *G1* to a second and third generation *G2* and *G3* is illustrated.

One big advantage of these methods is their relative independence with the number of variables and the complexity of the objective function. Though, they require a very large number of evaluations before the optimisation can be deemed satisfactory, and have a tendency to find local extremum only.

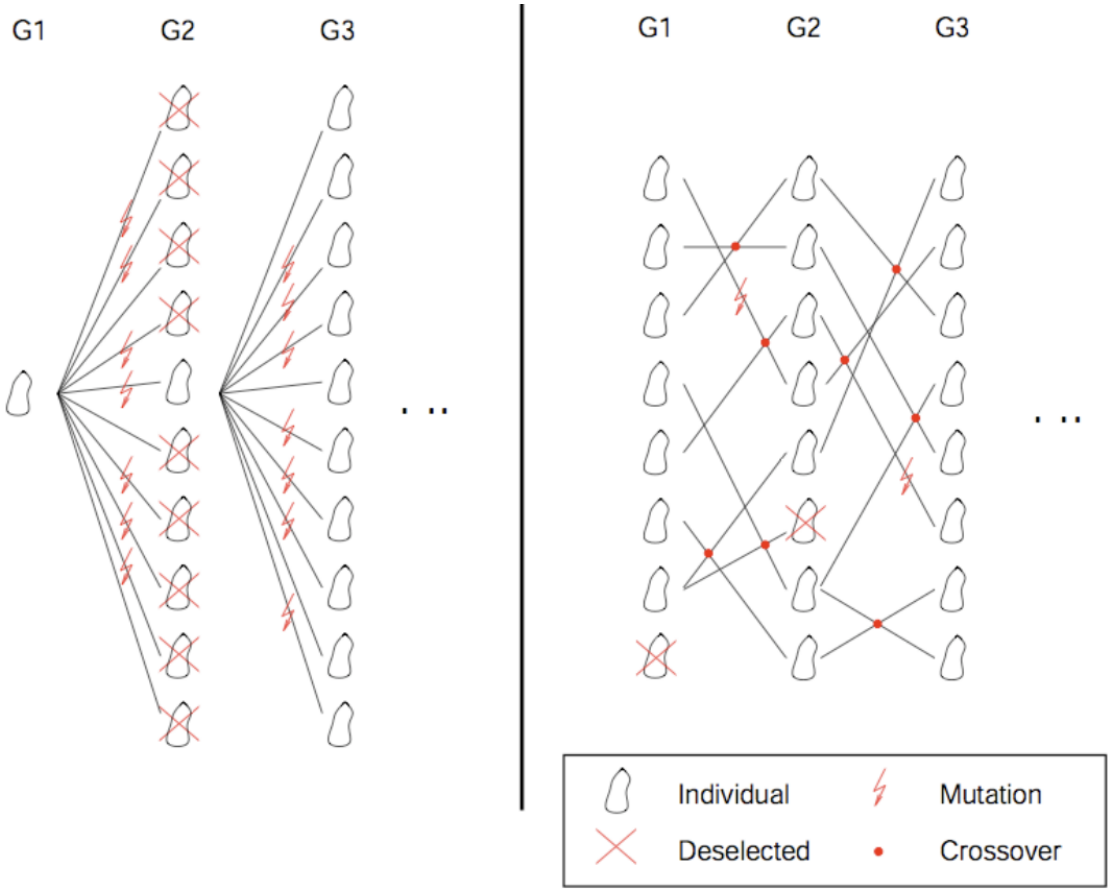


FIG. 2.5 – Mutation (*left*) versus re-combination (*right*) mechanisms (*Source: [38]*).

2.3.3 Meta-heuristic methods and other nature-inspired algorithms

Meta-heuristic methods gather optimisation methods which successively try to improve the design based on the knowledge already gained by the algorithm. A lot of them are nature-inspired algorithms such as the methods presented in [26].
The first category of meta-heuristic methods is *population based*. The design population will evolve through a parallel search depending on the performance of previous populations ; in the same manner as the Evolutionary Algorithms. Amongst these, Particle Swarm Optimisation ([69]), Ant Colony method ([33]), Bee Colony Method ([117]), Hunting Search ([93]) and Bacterial Foraging ([20]) can be mentioned.
The other category of meta-heuristic methods or *trajectory methods* explores the design through a single evaluation of the objective and constraint functions at each iteration. The mechanisms to evolve from an iteration to the next one are then pretty

diverse. Simulated Annealing ([71]), Tabu Search ([52]), Harmony Search ([51]) and Hybrid Cellular Automata ([118]) can be mentioned. This last method is, unlike the other methods, an element-based optimisation method and will be detailed in Section 3.5.

2.3.4 Methods using meta-models

Optimisation methods using meta-models rely on building a simplified version of the objective and constraint functions on which a given optimisation method will be applied. This method is largely discussed in reference books such as [89, 47]. Usually, the meta-model is an analytical expression (named *Response Surface*, see for instance Figure 2.6 where a quadratic response surface based on two parameters is illustrated) approximating the objective and constraint functions based on a finite number of evaluations of these functions.

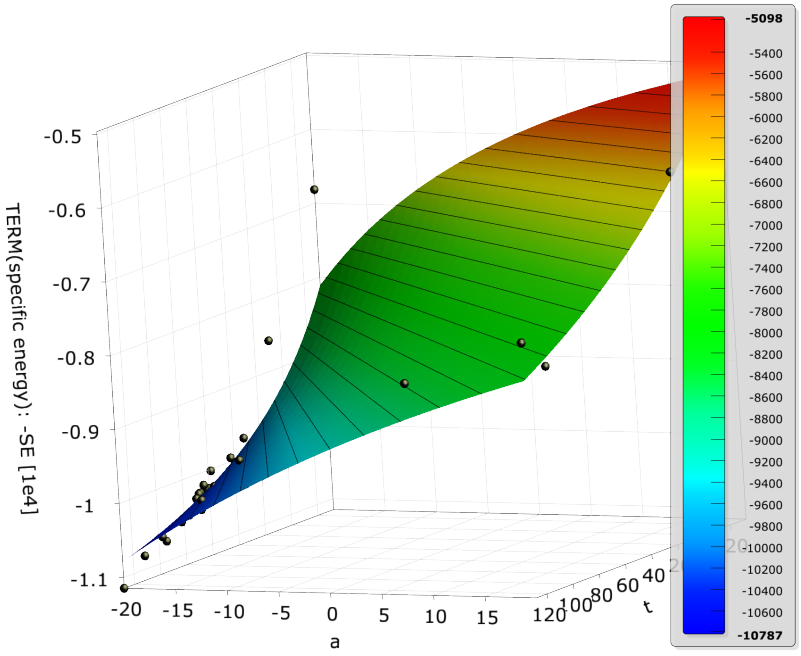


FIG. 2.6 – Example of Response Surface: Specific Energy (-SE, 1e4 J/kg) function of shape parameter a (mm) and thickness parameter t (1e1 μm) (generated from an optimisation problem presented in [36], see also case A on Figure 2.3). The response surface is generated using the support points (black dots). Because of the complexity of the response, the fit between the response surface and the support points is not perfect.

Practically, the optimiser defines a set of design points for which the crash analysis will be performed and the objective and constraint functions extracted. This set can be generated with stochastic methods (Monte Carlo, Latin Hypercube) or through deterministic space filling methods (linear, D-optimal, quadratic...).
Depending on the number of design variables, and on the computational resources available, the Response Surface can be built after a linear, quadratic, or higher

order polynomial. Other methods such as Radial Basis Functions (RBF, see [21], often related to the Neural Networks), Kriging (developed after [73]) or Support Vector Regression (SVR, see [34]) can also be used to generate the Response Surface. The number of crash analysis necessary to build the Response Surface is a direct consequence of this choice.

Once the Response Surface has been built, a classical optimisation method (for example gradient-based methods or Evolutionary Algorithms coupled with gradient information) is used to find the optimum for the meta-model of the function. A crash analysis should be done then to check that the objective and constraint values of the optimum found on the Response Surface are accurate.

For crashworthiness design, as the objective and constraint functions are normally highly nonlinear, the Response Surface can be inaccurate. In this case, iterative Response Surface methods can be used. These methods generate successive Response Surfaces, gradually reducing the design space to focus on the area of interest and improve the quality of the Response Surface around the optimum. Such methods have been used in a number of crashworthiness applications such as [59, 104, 83].

Chapter 3

Topology Optimisation for Crashworthiness Design

While Chapter 2 introduced optimisation in the context of crashworthiness design, this chapter explores the more specific area of topology optimisation for crashworthiness design. Here, the more general definition of topology optimisation is taken (*i.e.* deriving the best structural topology within a given design space).
Structural optimisation for crashworthiness design has already been well documented, but most applications only deal with size optimisation (see for instance [104, 35, 56]). Applications to shape optimisation also start to emerge with for instance works such as [11, 128, 83, 103, 59]. However, topology optimisation applications remain really limited even though their potential to improve structural designs is much bigger.
Topology optimisation is well established for static mechanics or even electromechanical problems. It has been extensively documented in a reference book by Bendsøe and Sigmund [16] and is still today the main topic of publications in the scientific journal *Structural and Multidisciplinary Optimization*[1]. Classic topology optimisation methods such as Solid Isotropic Material with Penalisation (SIMP) [14] rely on node-based sensitivities to perform the optimisation (see Section 2.2) and the number of design variables is equal to the number of finite elements in the initial model. This is not adapted to crashworthiness design with explicit finite element simulation as computing sensitivities throughout the model would require an unrealistic amount of computations which are already expensive on their own. Alternative methods have been developed based on two principles. Firstly, the explicit finite element simulation can be replaced by a less expensive simulation method (see Section 1.2.3). Secondly, the classic topology optimisation method can be replaced by an alternative optimisation method usually reducing the number of design variables or a method which does not guarantee the optimality of its result.
In this chapter, the main methods used so far for topology optimisation in crash-

[1] `www.springer.com/materials/mechanics/journal/158`

worthiness design are presented in Sections 3.1 to 3.5. The advantages and drawbacks of each method are discussed in Section 3.6 and the need for an alternative method is demonstrated at the end of this chapter.

3.1 Equivalent Static Loads (ESL) based methods

The ESL methods are based on the fact that, under certain conditions, a fully dynamic non-linear crash simulation can be replaced by a series of consecutive linear simulations using static loads with the same deformation effects as the dynamic load. Those static loads are referred to as the Equivalent Static Loads. With a relevant discretisation of the transient phenomenon, the computation time of the ESL simulation can be much lower than the explicit simulation as only linear simulations need to be ran during the optimisation process.
In any case, the method needs an initial explicit simulation to gauge the static load cases, which is why it is not mentioned in Section 1.2.3 as an alternative simulation method to the explicit simulation. Nevertheless, it can be used within an optimisation process to accelerate the evaluation of design variables changes. The method has been applied for size (see [95]) and shape optimisation (see [122]), but this section focuses on the application to topology optimisation.
Two different forms of ESL methods should be distinguished, as described in Section 3.1.1 and Section 3.1.2.

3.1.1 ESL and global static loads

As mentioned previously, this method uses linear simulations to compute the responses to equivalent static loads ; the optimisation is then performed with respect to these linear load cases. This section focuses on global ESL representing the interactions between the vehicle or component to optimise and the external elements involved in the crash.
Volz [122, 37] presented a version of the method where the loads are defined using crash kinematics considerations. The optimisation of the rear part of a vehicle under high velocity rear crash conditions is given as an example. First the ratios of energy which should be absorbed by the different areas of the design domain are defined. This is done taking into consideration other load cases such as low velocity impacts. Combining these different energy absorption targets and geometric considerations, force levels are derived with their time dependency. From these force levels, the ESL are defined and the multi load case static linear topology optimisation can be performed using classic methods.
Christensen presented a different approach of the ESL in [28]. In this case, a single static load is applied to represent the crash load. This can be seen as an extreme ESL where only one time step is taken. The static topology optimisation problem is

then solved using the SIMP method [14]. In this method, the behaviour of each finite element is interpolated from the initial material. The interpolation is based on the current density of the element and a penalisation factor p. This penalisation method drives the solution to 0-1 values for the density of each element and allows therefore to define regions of material or no material within the initial design domain.
The work presented by Cavazzuti *et al.* [25] should also be mentioned. In this paper, they performed a topology optimisation of a full vehicle structure (see Figure 3.1) under different load cases. A crash load case was included where the dynamic explicit simulation was replaced by a single static load case applied to the wheel centres, the engine, the gearbox and seat joints. Again, the transient nature of the crash is not represented here.

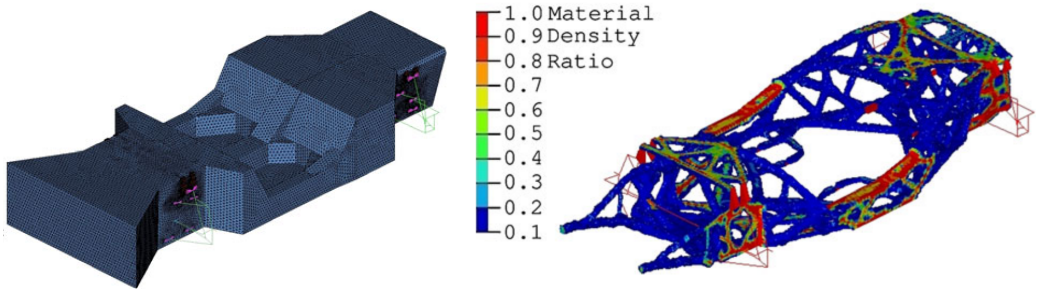


FIG. 3.1 – Topology optimisation domain (*left*) and result (*right*) for a full vehicle structure using a global equivalent static load (*Source: [25]*).

3.1.2 Nodal based Equivalent Static Loads

This method was introduced by Kang *et al.* [68] and Choi *et al.* [27] and further developed by Park *et al.* [110, 95, 127]. It is also implemented in the commercial softwares GENESIS [120] or OptiStruct [6]. Compared to the previous method, the Equivalent Static Loads are now defined at the finite element level, *i.e.* artificial forces are applied to each node of the model. This allows for a finer control of the model deformations but the effort to extract the loads is higher.
Formally, the Equivalent Static Loads are derived as follows. Taking into account the design variable $\mathbf{x}$ dependency, Equation (1.7) can be re-formulated as:

$$\mathbf{M}\left(\mathbf{x},\, t^n\right)\ddot{\mathbf{d}}^n + \mathbf{K}_{\mathrm{NL}}^n\left(\mathbf{x},\, \mathbf{d}^n,\, t^n\right)\mathbf{d}^n = \mathbf{f}^{\mathrm{ext}}\left(\mathbf{d}^n,\, t^n\right). \tag{3.1}$$

Where $\mathbf{K}_{\mathrm{NL}}^n$ is the non-linear stiffness matrix of the model at time step n. Or even:

$$\mathbf{M}\left(\mathbf{x},\, t^n\right)\ddot{\mathbf{d}}^n + \mathbf{C}^n\left(\mathbf{x},\, \mathbf{d}^n,\, t^n\right)\dot{\mathbf{d}}^n + \mathbf{K}_{\mathrm{NL}}^n\left(\mathbf{x},\, \mathbf{d}^n,\, t^n\right)\mathbf{d}^n = \mathbf{f}^{\mathrm{ext}}\left(\mathbf{d}^n,\, t^n\right). \tag{3.2}$$

Where $\mathbf{C}^n$ is the damping matrix of the model at time step n.
The Equivalent Static Load at time step n, $\mathbf{f}_{\mathrm{eq}}^n$, is then defined as the load verifying:

$$\mathbf{f}_{\mathrm{eq}}^n = \mathbf{K}_{\mathrm{L}}\left(\mathbf{x}\right)\mathbf{d}^n. \tag{3.3}$$

Where $\mathbf{K}_{\mathrm{L}}$ is the linear stiffness matrix of the model.
Using a set of Equivalent Static Loads for each desired time step, the topology optimisation problem can be solved as a multi load case linear static topology optimisation problem.
An overview of the applications of these methods is given in [96], including both size and topology optimisation problems. Other applications were presented by Gambling [50] to optimise the topology of a bumper impacted by a pole, or by Erhart [39] to optimise the cross-section of a rocker in the event of pole impact.

3.2 Ground Structure approach (GSA)

The Ground Structure Approach consists in filling the design space with elementary macro-elements with a simplified crash behaviour and using different methods to remove and/or modify these macro-elements to reach an optimum design. In 1999, Soto and Diaz [112] proposed to use a lattice of truss elements to fill the initial design space. These elements are given a non-linear strain-displacement behaviour and a strength parameter. By modifying this strength parameter, one will be able to optimise the structure. In 2002, Pedersen [98] proposed a ground structure approach composed of rectangular 2D beam elements with plastic hinges. Those elements can undergo large rotations and their main design variable is their cross-section dimension. A simplified application example by Pedersen is illustrated in Figure 3.2.

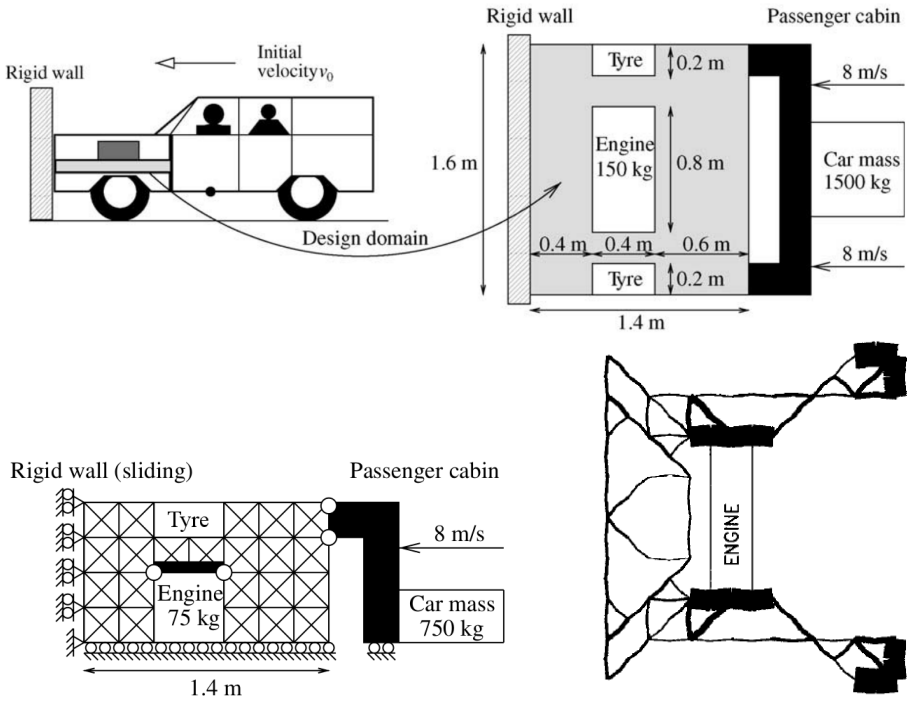


FIG. 3.2 – Ground Structure approach application, load case and design space (*top*), ground structure (*bottom left*) and optimised structure (*bottom right*) (*Source: [99]*).

The method was further expanded by Fredricson [49]. He introduced a Ground Structure made of beams and joints although in this case, only static load cases were studied. A first application example was the classic topology optimisation of a rectangular plate clamped on one side and loaded vertically at a corner of the opposite side. A second one was the optimisation of a full vehicle structure for three different load cases, one bending case and two twisting cases.

3.3 Graph and heuristic based method

The Graph and heuristic based method is an approach developed by Schumacher *et al.*. It was first introduced in [107] and is extensively detailed in [94]. Somehow, it can be seen as an extension of the earlier Bubble Method also developed by Schumacher *et al.* (see [106]).

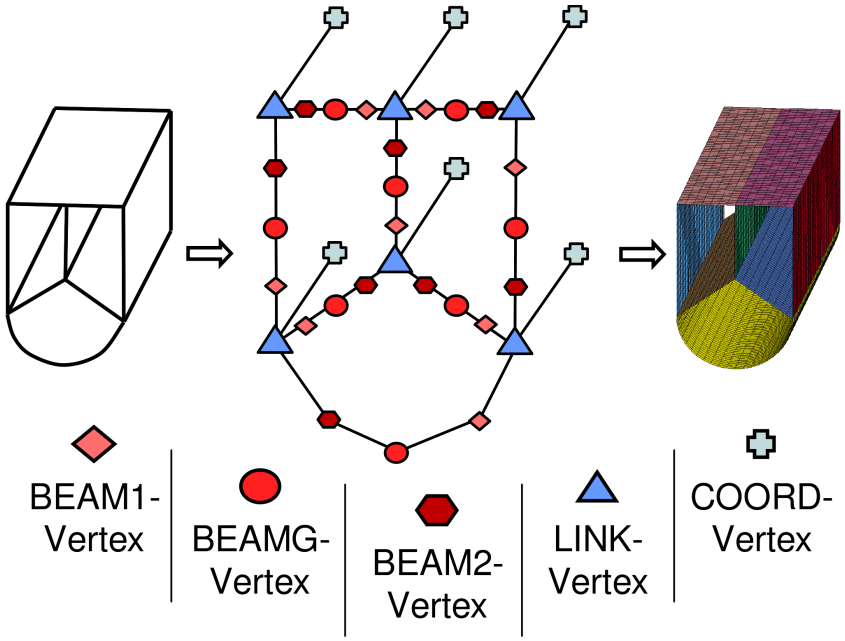


FIG. 3.3 – Graph representation of a longitudinal structural component (*Source: [94]*).

This method is based on an abstract graph representation of the structure, usually for 2D design spaces (such as the cross-section of longitudinal elements). Figure 3.3 illustrates this representation of structural components with an abstract graph. The topology optimisation method is divided into two processes. An outer one where the topology level of the cross-section is iterated and an inner one where a shape optimisation is performed on the current topology level. The shape optimisation is performed using algorithms based on global outputs (see Section 2.2) and explicit finite element simulations of the crash. After each shape optimisation result, the topology level of the structure is modified by adding or removing walls in the cross-section. This modification is made using heuristic rules described in [94]. These rules evaluate the contribution of the different walls of the cross-section and for instance

delete the walls with low energy absorption or introduce new walls to support the walls with fast deformation. Each rule is given a priority value depending on the current configuration and the rule with highest priority is applied to define the new topology level. At the end of the shape optimisation process, if the design is not improved when compared to the previous topology, another heuristic rule with lower priority is applied on this previous topology. The optimisation stops when no heuristic rule manages to improve the design.

3.4 Bi-directional Evolutionary Structural Optimisation (BESO)

The Bi-directional Evolutionary Structural Optimisation (BESO) method is an extension of the Evolutionary Structural Optimisation (ESO) method where both addition and removal of material is allowed to derive the best topology. It was initially developed to solve static topology optimisation problems [102] but was later adapted to crash applications by Huang *et al.* [58]. More recently, Aulig [10] presented another application of the BESO method for crash.

In this method, the structure is designed to maximise the energy absorption efficiency (*i.e.* the energy absorbed divided by the mass of the structure) while respecting some force and displacement constraints. After each finite element simulation, the elements of the model are assigned two sensitivity numbers. Those sensitivity numbers should allow to optimise for two different criteria, the energy absorbed per unit volume e_1 and the ratio of energy absorbed over the ideal energy absorption e_2. e_1 is defined as:

$$e_1 = \frac{W}{\mathcal{V}} \tag{3.4}$$

Where W is the external work and $\mathcal{V}$ the volume of the structure. e_2 is defined as:

$$e_2 = \frac{W}{W_{\max}} \tag{3.5}$$

Where $W_{\max} = \mathrm{f}_{\max}\mathrm{d}_{\max}$ is the ideal energy absorption and $\mathrm{f}_{\max}$, $\mathrm{d}_{\max}$ are the force and displacement constraints respectively. To ensure the respect of the displacement constraint, the displacement of the impactor is gradually increased to the constraint value and the equilibrium is checked at each step. The sensitivity numbers are either derived from the final displacement or from the whole displacement history. Depending on these sensitivity numbers, elements will be either added or removed from the model. Also, the volume fraction of active elements in the model is adjusted in order to respect the force constraint. This process is iterated until an optimum is found.

An application example of the method is illustrated in Figure 3.4. This application is detailed in [58]. A rectangular plate fixed at both ends is impacted by a rigid

object and should respect both a displacement and a force constraint. Here, the displacement is gradually increased to the constraint value of 20 mm. The design is optimised using the BESO method, using both criteria described previously.

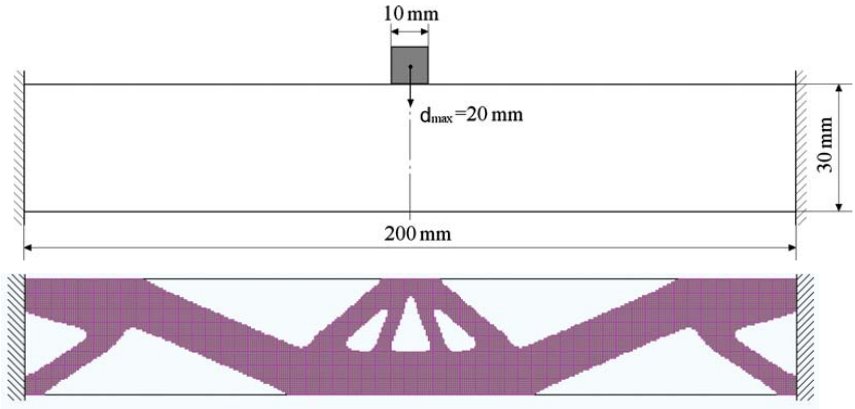


FIG. 3.4 – Application example of the BESO method. Design space and load case (*top*) and best topology derived (*bottom*) (*Source: [58]*).

3.5 Hybrid Cellular Automata (HCA)

Cellular Automata (CA) were first developed in the mid-20th century by Ulam and Von Neumann [119, 91]. It is a computational technique used to model different physical phenomena where the domain is represented by a grid of cells, each characterised by a finite number of possible states and evolving depending on the state of their neighbours. CA has been applied to structural optimization from the 1990s with the works of Inou *et al.* [61] and then Gurdal *et al.* [2, 116] or Kita *et al.* [72]. Hybrid Cellular Automata (HCA) for topology optimisation was defined by Tovar *et al.* [118, 100]. It is inspired by the biological process of bone remodelling. Like CA it uses a regular grid of cells, each characterised by a finite number of states. Through update rules it derives a structure where only relevant cells are kept. HCA differs from CA because it uses not only local information (cell and neighbourhood), but also global information (design space) within the update rules.

HCA has been adapted for the synthesis of topologies in crashworthiness design by Patel [97] and then Mozumder [87, 88] at University of Notre Dame (USA). They introduced explicit finite element simulation of the crash at each iteration of the HCA algorithm. The whole process is illustrated in Figure 3.5 and the notations are clarified in Table 3.1.

As a starting point the 3D design space is filled with solid finite elements. Each of the finite elements is a cell of the algorithm and is characterised by its material density level (or thickness for shell elements). Each cell has an equivalent role in the optimisation process. Mozumder also adapted the method for 2D design spaces

and shell elements (see [87] and Figure 3.6). In this case the HCA method becomes a topometry optimisation method as the shell elements are characterised by their thickness.

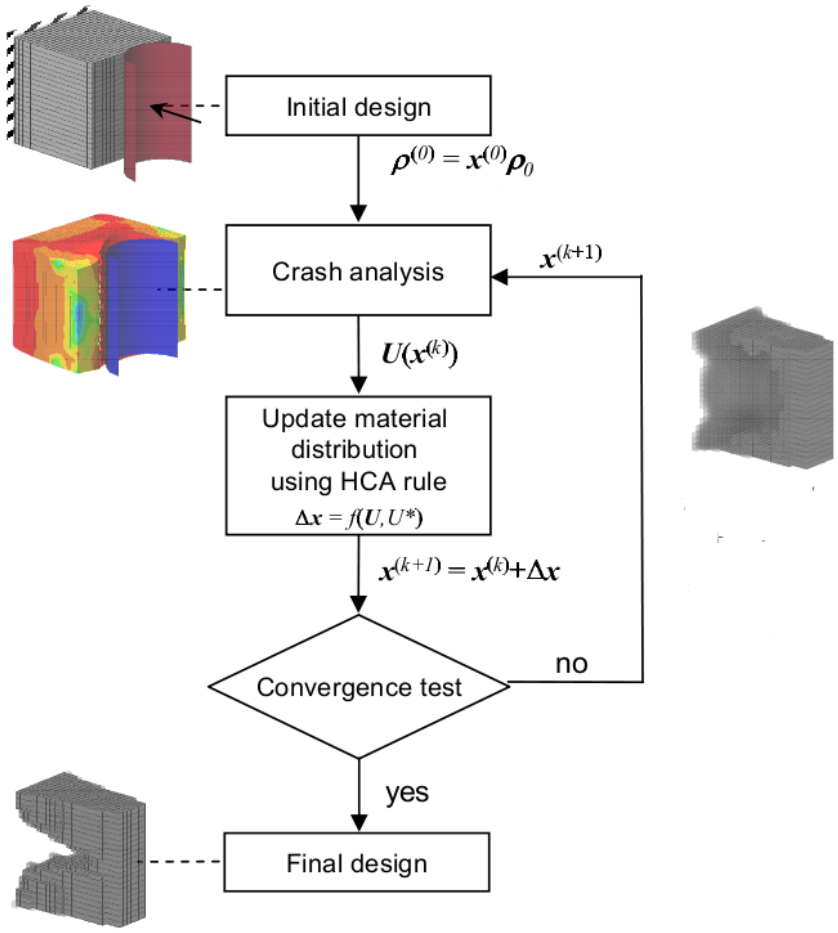


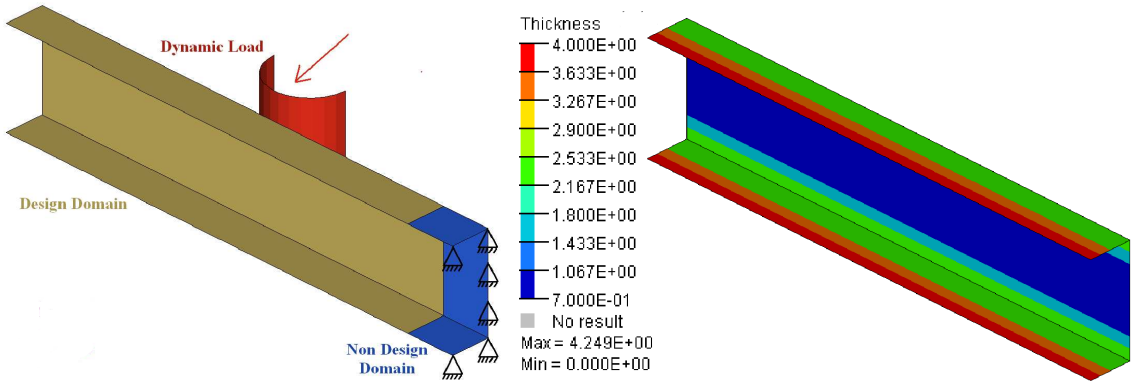
FIG. 3.5 – Hybrid Cellular Automata process for crashworthiness design (*Source: [97]*).

variable	definition
$\boldsymbol{\rho}^{(k)}$	material density distribution, iteration k
ρ_0	reference material density
$\mathbf{x}^{(k)}$	design variables vector, iteration k
$\boldsymbol{U}\left(\mathbf{x}^{(k)}\right)$	IED distribution, iteration k
$\Delta\mathbf{x}$	design variable change
U^*	IED setpoint

TAB. 3.1 – Hybrid Cellular Automata variables.

At each step of the algorithm, the current design is evaluated for crashworthiness through explicit finite element computation. The driving output of the computation is the Internal Energy Density (IED) of each cell (both elastic and plastic deformations). Each cell is supposed to meet a specified level of IED called *Setpoint*. Given the difference between the *Setpoint* and the output of the simulation, an update rule modifies the material density of each cell for the next step. If the IED is lower than

expected, the density of the cell is decreased, if it is higher than expected, the density is increased. Also, under a specified minimum density level, the cells are simply erased from the finite element model. Therefore, the topology of the structure can be continuously changed during the process. Notice that the definition of the minimum density level is not straightforward. With a really low level, the model will contain at some point elements with low density and therefore really low stiffness. These elements can be unstable for crash simulations. However, if the minimum density level is too high, the difference between the model behaviour with and without deleted cells can be too important. Theoretically, the update rule is meant to derive a homogeneous level of IED throughout the cells. When this happens, the algorithm stops, the structure is defined.



(a) Design space and load case.

(b) Best thickness distribution (mm) with constrained thickness along length.

FIG. 3.6 – Application example of HCA method for a 2D design space (*Source: [87]*).

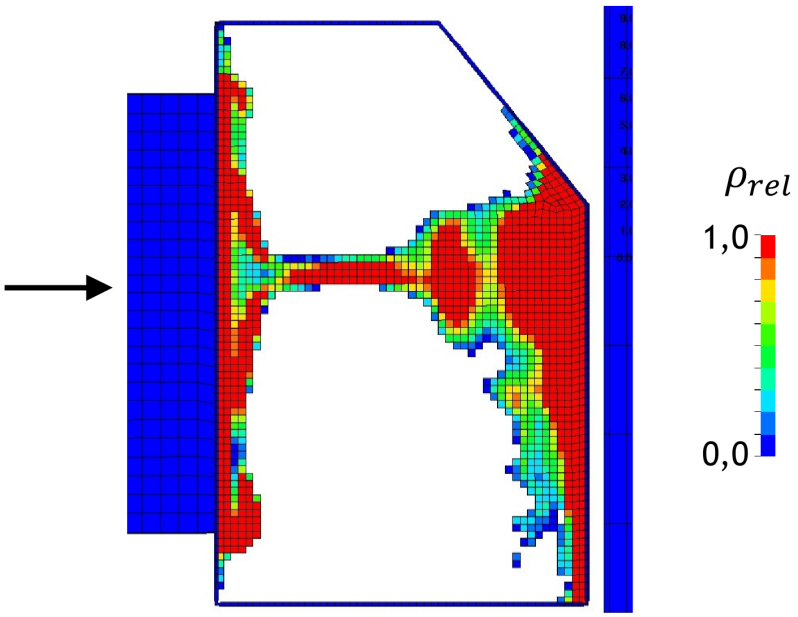


FIG. 3.7 – Application example of Hybrid Cellular Automata for crashworthiness design of a longitudinal component under pole impact. Material density distribution for best design (*Source: [39]*).

The method has been implemented in the commercial software LS-TaSC [79] and an application example using this software is given in [39] (see Figure 3.7). A slightly modified version of the algorithm was also presented by Guo *et al.* [53] where the IED criterion is enriched with a constraint based on the strain levels of the cells. This modification allows to control the maximum strain level in the structure.

3.6 Discussion of the available methods

The methods presented in Section 3.1 to 3.5 represent a state-of-the-art of topology optimisation methods for crashworthiness design. Some alternative methods were not presented as they were deemed less significant or somehow similar to the listed methods. The work presented by Forsberg [48], for instance, can be linked to the HCA and BESO methods for its use of the Internal Energy Density to characterise efficient structures.

Discussions and comparisons of these available methods can already be found in literature. However, given the nature of each method, it can be difficult to compare them directly as they do not necessarily offer the same design objectives and constraints capabilities. In [39], HCA and ESL are compared on two different topology optimisation problems of vehicle components. In [29], global ESL and HCA are compared on a full vehicle topology optimisation problem.

To simplify discussion, the topology optimisation methods are divided into two categories. On the first hand, methods such as node-based ESL, BESO and HCA which fill the design space with solid elements and derive bulk topologies. On the other hand, GSA and Graph and heuristic based method which derive thin wall structures exploring only a subset of possible topologies.

As the first type of methods derives bulky structures, they are probably better suited for design tasks where a high fraction of the initial design space should be filled with material. To derive structures with a low fraction of the initial design space, the discretisation should be really fine which strongly penalises the computation time. In [39], an example application of the HCA method is presented. The design task is to optimise the topology of a longitudinal member for pole impact. As shown in Figure 3.7, the discretisation used is very fine. In this case, ca. one million elements are used to mesh a single car component. When compared to the usual finite element models of complete vehicles around 2 million elements, it is clear that the computation cost necessary here is not affordable.

As discussed in [77] and [29] this first type of methods usually involves some post-processing after the topology optimisation to transform the bulk structure into a thin-walled structure, more suitable for manufacturing. This process is not straightforward. Whether it is done manually or with automatic geometrical processing, the interpreted structure can behave differently from the topology optimisation result.

Further shape or size optimisation can be performed on the interpreted structure, but this means more parameterisation and computation time.
The node-based ESL method presents an additional difficulty. Since the ESL are node based, modifications of the finite element mesh are not possible during the static optimisation process. As a result, elements with really low material density cannot be deleted but can still have an important influence on the structure behaviour.
On the other hand, methods deriving thin-walled structures are limited by the fact that they do not explore the full design space. In the GSA, the initial Ground Structure defines the subset of possible solutions for the optimisation process. Similarly, the Graph and heuristic based method only explores a finite number of possible topology classes. In both cases, a subspace of the design space is investigated.
A specific limitation of the Graph and heuristic based method is that, so far, it is only able to define the topology of cross-sections, in other words, it can only solve problems with a 2D design space (the structures investigated can be 3D though). Also, the definition of the heuristics to modify the topology (see [94]), is strongly biased towards lateral impacts. Applying the method to other load cases would probably involve defining different heuristics. Eventually, the two examples presented in [94] both needed several thousands explicit simulations which is too much for industrial applications.
As topology optimisation applied to crashworthiness is still in its early stage, it would be premature to completely dismiss a given method. As suggested in [37], different methods may be necessary for the different fields of topology optimisation. Yet, it is clear that another method can be developed to try and get past some of the shortcomings described here. Chapter 4 introduces an alternative method for topology optimisation in crashworthiness which tries to use the benefits of the different methods presented in Chapter 3.

Chapter 4

Hybrid Cellular Automata for Thin-walled Structures

In Chapter 3 the interest of developing an alternative method for topology optimisation in crashworthiness design was demonstrated. Early applications of the HCA method showed that it has some advantages as it uses more realistic explicit crash simulation. Also, the use of local output information (in the form of the cells' internal energy density) gives some kind of sensitivity of the structure to the crash response. As discussed in Section 3.6, one of the limits of this method is the use of solid elements. Even though Mozumder already adapted the HCA method to thin-walled structures [87], this was only for topometry optimisation of 2D structures (see Figure 3.6). The method presented in this chapter proposes to generate 3D topologies with thin-walled structures. Also, for easier post-processing and to reduce the number of design variables in the problem, the finite element cells of the classic HCA are replaced by macro element cells gathering several finite elements.
In Section 4.1 of this chapter, the use of thin-walled structures for energy absorption is discussed. This is ground for the development of the topology optimisation method presented in Section 4.2. In that section, the difference between this method and the classic HCA is discussed. In Section 4.3, the method's algorithm is detailed. Eventually, the practical implementation of the method is presented in Section 4.4.

4.1 Thin-walled structures for energy absorption

As discussed in Section 1.1.3, lightweight design is especially important for automotive crashworthiness. To enable automotive lightweight design, the use of thin-walled structures for energy absorption has been crucial.
Because of their specific deformation characteristics (see for instance [3, 70], Figure 4.1 or Figure 4.2 and many other publications in the journal *Thin-Walled Structures*[1]), they usually provide better energy absorption density than plain structures.

[1] `www.journals.elsevier.com/thin-walled-structures`

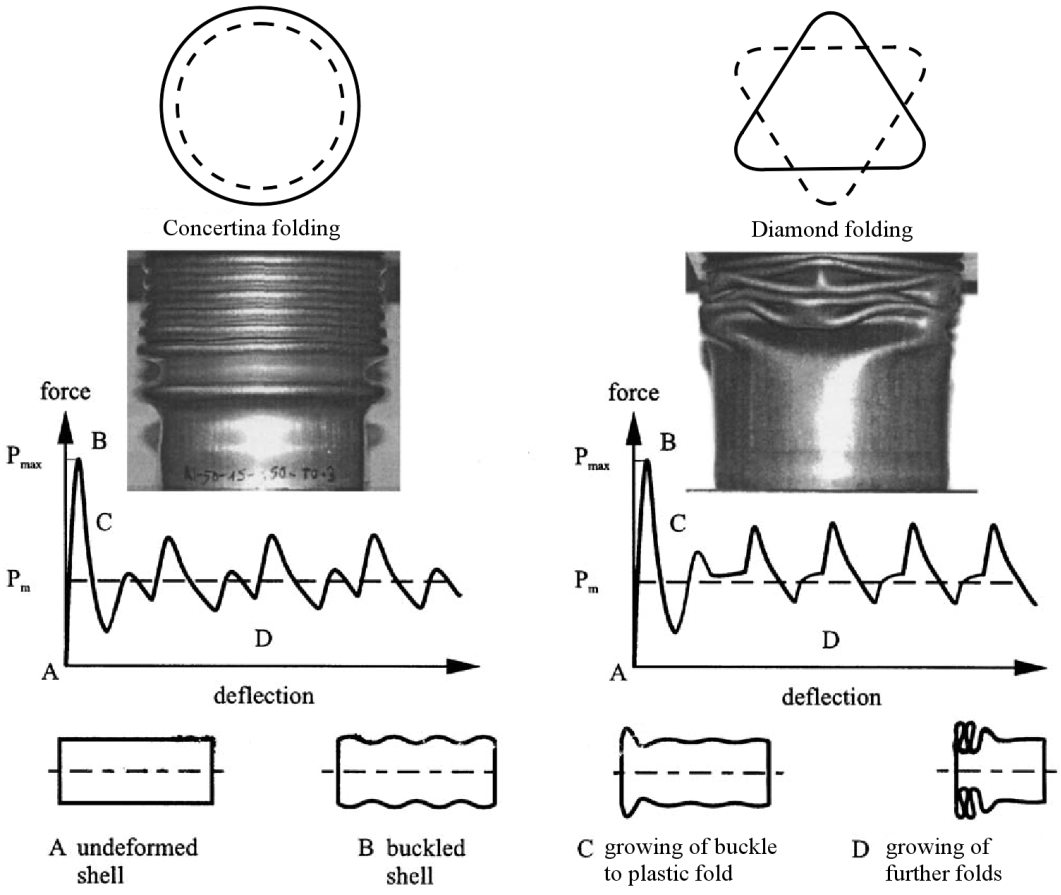


FIG. 4.1 – Concertina (*left*) and diamond (*right*) folding modes for cylindrical tubes after axial impact (*Source: [86]*).

FIG. 4.2 – Thin-walled tubes folding modes after axial impact (*Source: [3]*).

Figure 4.3 exhibits a very specific deformation mode with an inversion of the tube. Such a deformation mode is difficult to model with solid elements as used in the classic HCA or BESO. Indeed it requires using very small solid elements (in the range or smaller than the walls' thickness) which implies really long computation times.

The use of thin-walled structures for energy absorption is not limited to vehicle design. Applications to aerospace engineering are illustrated in Figure 4.4 and Figure 4.5 where the floor of an aircraft or an helicopter is reinforced with thin-walled structures.

FIG. 4.3 – Undeformed (*left*) and deformed (*right*) conic shells exhibiting an inverted mode of collapse (*Source: [54]*).

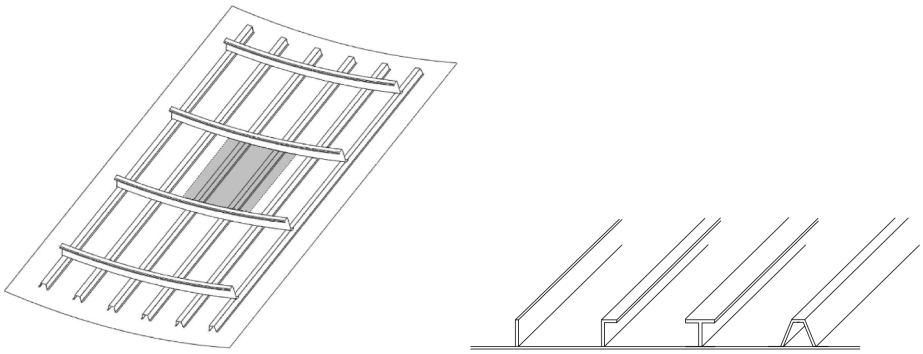


FIG. 4.4 – Composite panel (*left*) reinforced with thin stiffeners (*left and right*) (*Source: [121]*).

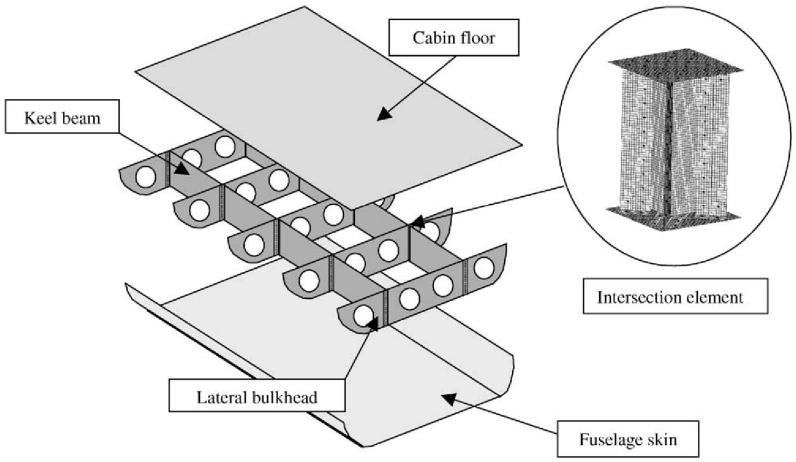


FIG. 4.5 – Typical helicopter subfloor structure for energy absorption (*Source: [17]*).

Most of the literature on thin-walled structures for energy absorption remains a simple analysis of the properties as in [86]. In some cases size optimisation (see for instance [104] with a vehicle crashworthiness application), or even shape optimisation is performed (see for instance [59] where shape optimisation of a prismatic beam's cross-section is performed). As mentioned in Section 3.5, Mozumder adapted

the HCA method to topometry optimisation of thin walled structures and Ortmann *et al.* [94] developed the Graph and heuristic based method (see Section 3.3) to derive the cross-sections' topology of energy absorbing structures. Yet, no method seems to be available to derive the topology of thin-walled structures in the case of crashworthiness design with a 3D design space. This is what the method presented in this chapter proposes to do.

4.2 Hybrid Cellular Automata for Thin-Walled Structures (HCATWS)

4.2.1 Deriving thin-walled structures with the HCA method

In order to derive the topology of thin-walled structures for crashworthiness design with a 3D design space, the method presented in this thesis proposes to apply the HCA method to a ground structure made of thin-walled macro elements. The use of the HCA method (see Section 3.5) allows to use explicit crash simulation in the optimisation process. The use of thin-walled structures in the optimisation allows to reproduce the specific deformation modes of these structures (see Section 4.1) and to model self contacts or wall-to-wall contacts more easily. It also enables to skip the process of interpretation necessary when optimising with solid structures but designing for thin-walled structures. This process is illustrated for instance by Cavazutti *et al.* [25]. In this paper, a topology optimisation is first performed on a full vehicle with solid elements (see Figure 3.1). Then, the result of this optimisation is manually interpreted to generate a thin-walled structure. This interpretation step is crucial as it is difficult to get the same structure behaviour with solid elements and shell elements. The interpreted structure is used to perform topometry optimisation and the result of this optimisation (see Figure 4.6, *left*) is interpreted once more before a sizing optimisation is performed (see Figure 4.6, *right*).

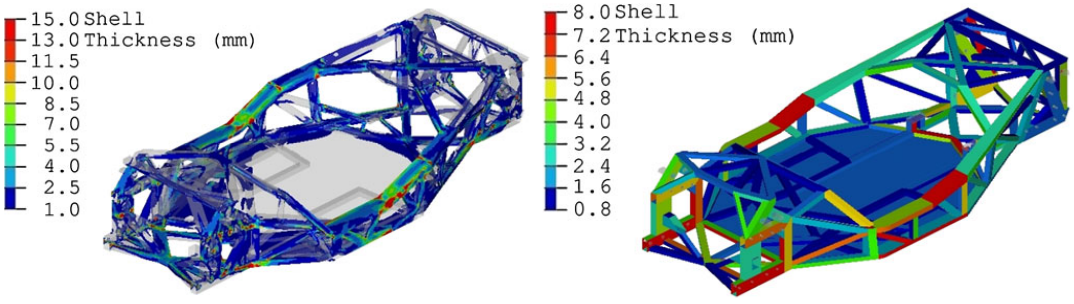


FIG. 4.6 – Thin-walled structure derived from a topology optimisation result (see Figure 3.1) and used to perform successively topometry optimisation (*left*) and then size optimisation (*right*) (*Source: [25]*).

The method presented in this chapter also differs from the classic HCA as it replaces the finite elements as cells of the HCA with macro elements. The cells are now small

thin walls made of a number of shell finite elements. This is necessary with thin walls as the deformation modes exhibited in Section 4.1 do not produce homogeneous energy absorption for all shell elements. Higher energy absorption can be observed along the plastic hinge lines. Also, using macro elements ensures that within one cell, folding of the wall is possible.

Design variables and related output value

As the solid finite elements cells of the classic HCA are replaced by macro thin walls, the design variables of the algorithm are no longer the density of the finite elements but the thickness of the walls. Note that in the HCA version developed by Mozumder [87], the design variables where the thickness values for each shell element of the model as opposed to the thickness of the macro walls here. As in the classic HCA described by Patel [97] and Mozumder [87], the IED (elastic and plastic deformation) of each cell is the corresponding output value. The algorithm will therefore try to homogenise the IED throughout the cells.

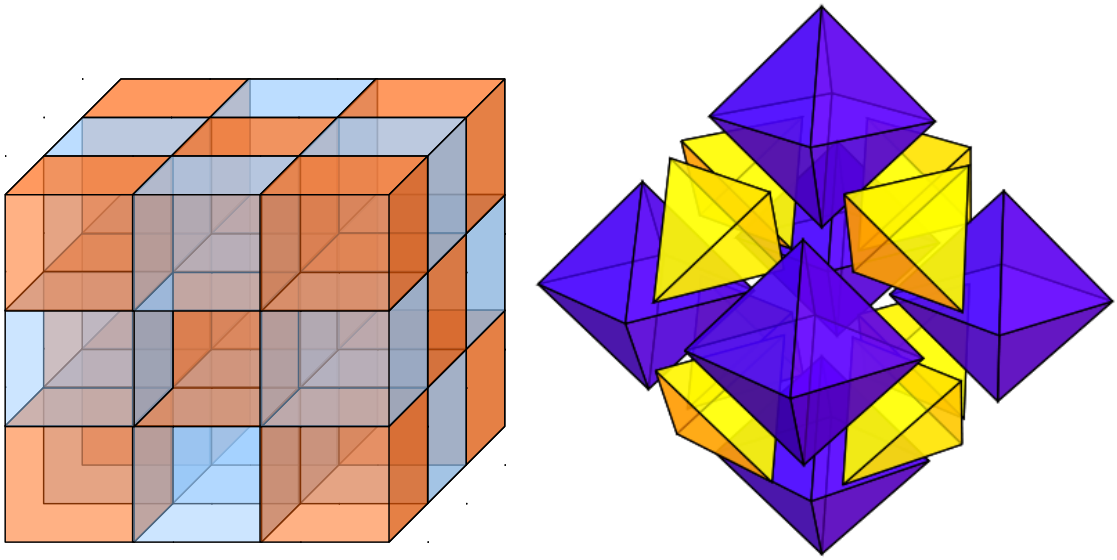
4.2.2 Space filling with thin walls

Geometry

The question of space filling is crucial for HCATWS. A 3D design space filled with thin walls is defined in geometry as a honeycomb[2]. As mentioned in Section 3.5, the local update rules of HCA suppose that the design space is made of a regular grid of cells. A non regular grid would mean the need for a non regular update rule amongst the cells which is not desirable. The only completely regular honeycomb or space filling in 3D is the cubic honeycomb [31] (see Figure 4.7(a)). It is therefore the space filling chosen for the first studies of the HCATWS method. Two quasiregular space fillings also exist with the tetrahedral-octahedral honeycomb (see Figure 4.7(b)) and the gyrated tetrahedral-octahedral honeycomb. Although they are not deemed regular in the space filling sense, they are face-transitive. As the faces of the honeycomb are the cells of the HCATWS grid, these honeycombs would generate a regular grid of cells. Still, they are not studied here as they do not allow to generate simple structures as easily as the cubic honeycomb. Another face-transitive (or isohedral) space filling to mention is the rhombic dodecahedral honeycomb. Because of its complexity, this honeycomb is not studied here. Prismatic honeycombs which are regular in 2D only are discussed in Chapter 5.

For a given design space, the number of cells to include in each direction must be chosen carefully. As mentioned previously, the size of the cells must be big enough to allow the apparition of folding deformation modes. Also, the cells should be big enough to allow defining structures which could be easily manufactured. Eventually,

[2]The question of 2D space filling with thin walls is treated in Chapter 5 and Chapter 6



(a) Cubic honeycomb.

(b) Tetrahedral-octahedral honeycomb, exploded view (*Source: [124]*).

FIG. 4.7 – Regular and quasiregular space fillings.

in a real design case the design space would not be cubic, therefore the number of boxes would be different in each (x,y,z) direction. The overall dimensions of the design space would probably help to define the dimensions of the nominal boxes.
As an illustration example, a cubic design space with a side of 300 mm is defined and filled with a cubic honeycomb space filling (see Figure 4.8). The design space is filled with $3 \times 3 \times 3$ cubic boxes with a side of 100 mm. Each face of the cubic boxes is considered as a thin-walled cell by the algorithm. To avoid repetitions of walls between adjacent cubic boxes, most of them are not made of 6 walls but 3, 4 or 5 walls. In this example, a mesh size of 5 mm is chosen, therefore each cell is made of $20 \times 20 = 400$ shell elements in the finite element model.

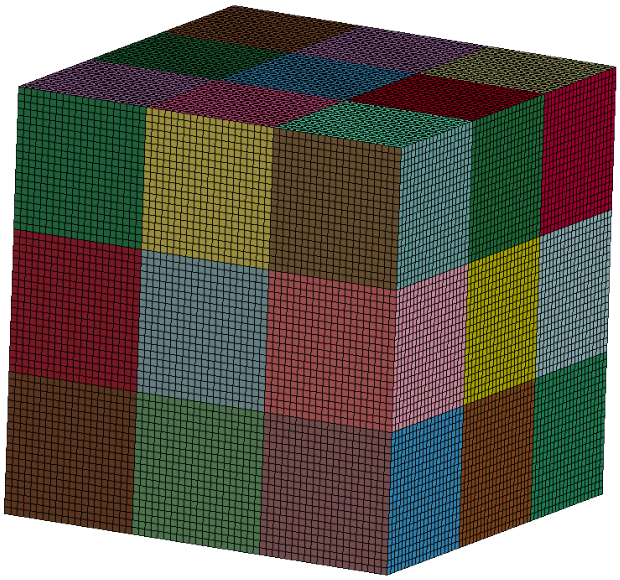


FIG. 4.8 – Illustration example of cubic space filling, finite element model.

Connections

To ensure structural interactions, the cells need to be connected to each other. The easiest way to create a connection between adjacent cells is to use shared finite element nodes along their common edge. Although this is really convenient, it is only relevant physically if the walls sharing nodes are manufactured in the same operation (*e.g.*: via extrusion or casting). In vehicle structures, the different panels of the structure are usually welded or glued together along connection surfaces called flanges. Therefore, depending on the design case, the cells can either share nodes along their common edges or have additional connection surfaces. Figure 4.9 illustrates these two types of connections.

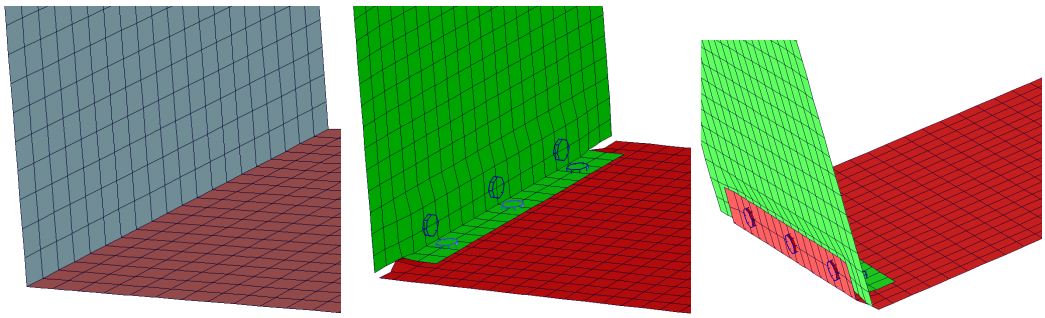


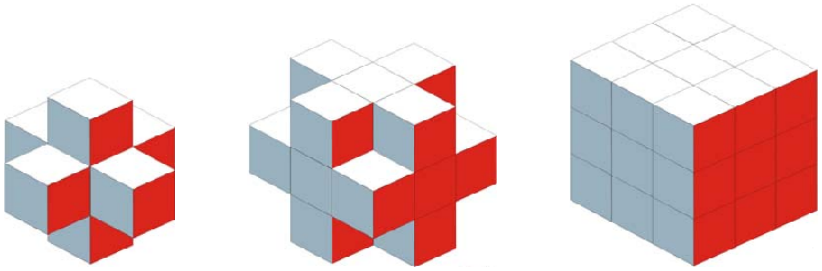
FIG. 4.9 – Connections between thin walls, shared nodes along connection edge (*left*) and flanges with spot welds (*centre and right*).

4.2.3 Neighbourhood

As mentioned in Section 3.5, in a Cellular Automata, the update rules for the cells use the output values of a given cell and its neighbouring cells. For the classic HCA implementations, this has a smoothening effect between the elements of the mesh and prevents the so-called chequerboard issue[3]. In the method presented here, each cell contains several finite elements, therefore it might not be necessary to use such a smoothening. Yet, the neighbourhood also gives information on how cells interact with each other and ensures continuity in the structure which is why neighbourhoods are used here too.

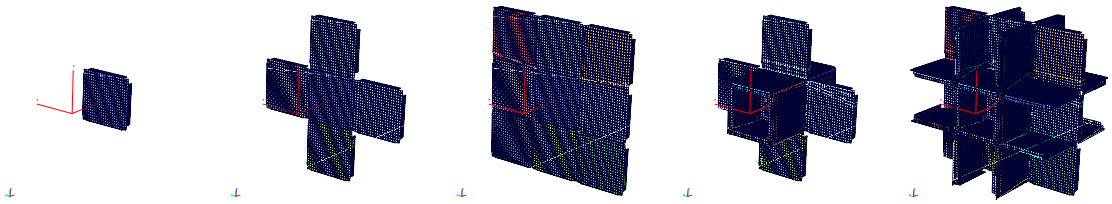
The extent of the neighbourhood influences the results of the HCA algorithm. Classic neighbourhoods for 3D solid structures are illustrated in Figure 4.10. The *von Neumann* neighbourhood gathers the elements sharing a face (the main feature for solid elements) with the central cell. In the *Radial* neighbourhood, the elements share edges. Eventually, in the *Moore* neighbourhood, the elements share nodes (the minor feature for solid elements). Defining neighbourhoods for thin macro elements

[3]When in the output of a topology optimisation the structure alternates, at mesh level, between material and void, it visually looks like a chequerboard (see [15]). While such structures generate good numerical results, they are not manufacturable and should therefore be avoided.



(a) Von Neumann (b) Radial (c) Moore

FIG. 4.10 – Classic neighbourhoods for 3D solid elements (*Source: [18]*).



(a) Empty (b) Plane von Neumann (c) Plane Moore (d) Von Neumann (e) Moore

FIG. 4.11 – Proposed neighbourhoods for thin macro elements.

is not necessarily trivial. Five different types of neighbourhoods are proposed in Figure 4.11. The *empty* neighbourhood (Figure 4.11(a)) is self-explanatory. In the *von Neumann* neighbourhood, the cells share edges (the main feature for thin walls), in the same plane (see Figure 4.11(b)) or in 3D (see Figure 4.11(d)). In the *Moore* neighbourhood, the cells share nodes (the minor feature for thin walls), in the same plane (see Figure 4.11(c)) or in 3D (see Figure 4.11(e)).
It is important to distinguish plane neighbourhoods (Figures, 4.11(b) and 4.11(c)) from 3D neighbourhoods (Figures 4.11(d) and 4.11(e)). During the crash simulation the loads will be transmitted differently between two cells belonging to the same plane and perpendicular cells. Using plane (or empty) neighbourhoods will favour a certain type of deformations and generate different topologies. In the case of non-empty neighbourhoods, different influences for the different types of neighbours can be assigned in the update rule. This allows to modulate the neighbourhood types.

4.2.4 Global performance objective and constraints

As detailed by Patel in [97], through the update rules, the HCA algorithm seeks the best material distribution to ensure a homogeneous IED distribution. To absorb energy efficiently, each cell of the structure should contribute to the global energy absorption equally. This concept is largely detailed in the different publications on HCA and therefore not detailed further here. For thin-walled applications, the homogeneous IED distribution criterion at the finite element level is questionable. Indeed, with the complex deformation modes described in Section 4.1, the IED distribution

is not necessarily homogeneous. However, with the macro cells used in HCATWS it is easier to reach the same IED level amongst the cells than amongst the finite elements as in the classic HCA approaches.
Patel [97] and Mozumder [87] defined different design constraints (maximum displacement, maximum reaction force, force displacement curve) for HCA in crashworthiness design. In most cases, these constraints are enforced by modifying the mass fraction target of the design space. The mass fraction is chosen as low as possible while respecting the design constraint. In other terms, the objective of the optimisation problem is to minimise the mass of the design while respecting a given constraint. The same approach is used in the HCATWS method.

Controlled deformation techniques

In some cases, it is desirable to finely tune the kinematics of the deformation. This can be done using different methods, with different levels of complexity. Without modifications to the algorithm, the user can divide the design space into different sub-spaces which should deform at different times during the crash. Then, instead of measuring the IED at the end of the crash, different acquisition times can be defined for the cells of each sub-space.
With a slight modification to the algorithm, the user can once again define sub-spaces and assign higher (respectively lower) IED targets to the sub-spaces of desired higher (respectively lower) energy absorption. This method, called graded IED, is already implemented in the classic HCA algorithm (see [87]). It can also be related to the method used by Volz in [122] to define its Equivalent Static Loads.

Static problems

The HCATWS method was developed specifically for crashworthiness design. Yet, the classic HCA method has been successfully applied to static problems (see [18]), so there is no theoretical contraindication in using HCATWS for static problems. Some static application examples are presented in Chapter 5 and Chapter 6. Ideally, HCATWS can be used to solve multi load cases problems including both crash and static.

4.2.5 Algorithm overview

The organisation of the new algorithm is largely inspired by the work of Patel [97] and Mozumder [87]. An outer loop modifies the geometry and finite element model according to the previous results and computes the crash simulation while an inner loop updates the thickness distribution until the mass target is satisfied. This organisation is detailed in Figure 4.12. Notations are clarified in Section 4.3. A nonlinear transient explicit simulation of the crash is performed on the initial design

or on the current design k. The IED level of each cell i, $U_i^{(k)}$ is extracted, along with the global output values $O_p^{(k)}$ (e.g. mass of the model, impactor displacement, force level). For each of these output values the corresponding constraint value is C_p. Each output should verify $O_p^{(k)} \leq C_p$ (for some outputs, there might be no constraint, therefore, $C_p = \infty$). Depending on the global outputs, the corresponding constraints and the previous mass fraction of the model, the new mass fraction objective $M_f^{(k)}$ is defined. The inner update loop is then activated to match the thickness distribution of the model with this mass fraction value. The IED setpoint $S^{*(j,k)}$ is defined at each step j of the inner loop. It is an energy absorption target for the cells, the algorithm will try to match the IED of each cell to this target. The setpoint depends on the current mass fraction objective $M_f^{(k)}$, the actual mass fraction derived from the thickness update at step $j-1$, $m^{(j-1)}$ and the setpoint at step $j-1$. Once the setpoint is defined, the thickness distribution can be updated using the update rules. These rules depend both on the current setpoint and on the IED output of each cell $U_i^{(k)}$. The inner loop is stopped once difference between the actual mass fraction and the mass fraction objective is lower than convergence parameter ε_1. Then, convergence of the outer loop is checked depending on the sum of thickness variations between iteration $k-1$ and k and convergence parameter ε_2. If convergence is not reached, a new finite element model is generated with updated cell thickness and added or removed cells using the CAE software SFE CONCEPT [109]. A new crash simulation is then performed.

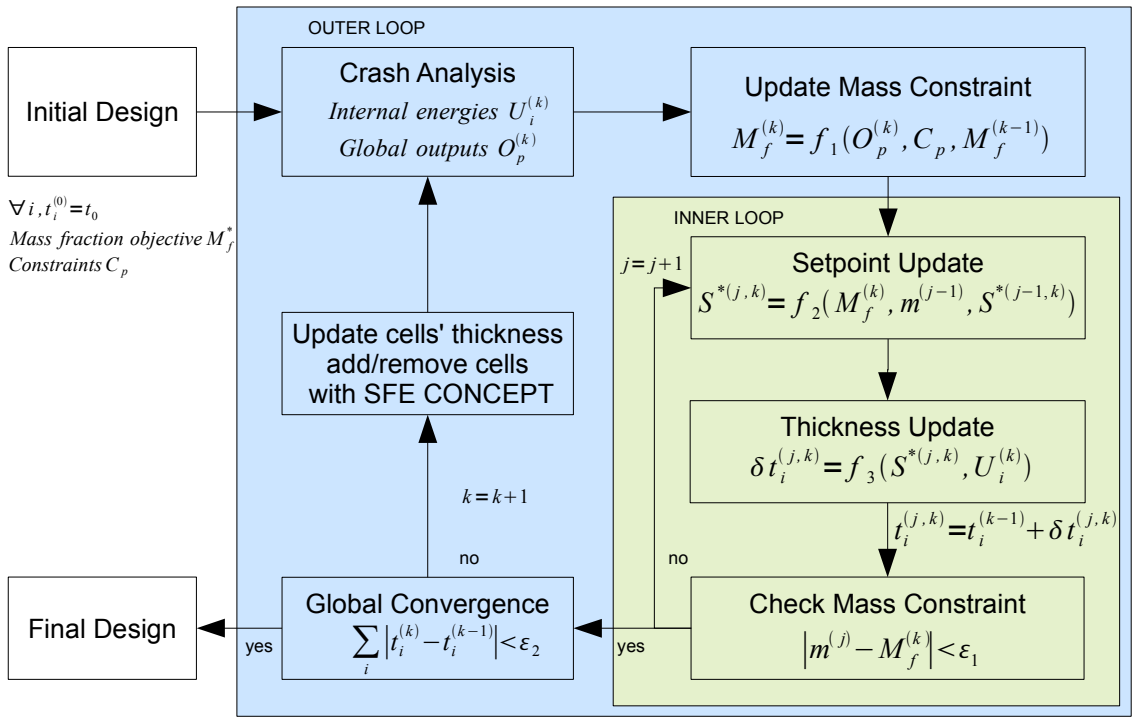


FIG. 4.12 – Algorithm overview.

4.3 Details of the algorithm

4.3.1 Mass constraint

The mass constraint is a result of the design constraint. Here, the maximum permissible displacement $\mathrm{d}_{\max}$ is the design constraint presented. As discussed in Section 4.2.4, other constraints can be implemented under the same principle. Given $\mathrm{d}_{\max}$ and the displacement output at iteration k, $\mathrm{d}_{\mathrm{out}}^{(k)}$, the displacement deviation $\varepsilon_{\mathrm{d}}^{(k)}$ is defined by:

$$\varepsilon_{\mathrm{d}}^{(k)} = \frac{\mathrm{d}_{\mathrm{out}}^{(k)} - \mathrm{d}_{\max}}{\mathrm{d}_{\max}}. \tag{4.1}$$

The mass fraction objective $M_f^{(k)}$ (defined as the mass objective divided by the reference mass) for iteration k is updated using the following equation:

$$M_f^{(k)} = \max\left(M_{f,\min},\ M_f^{(k-1)} + \Delta M_f^{(k)}\right). \tag{4.2}$$

Where $M_{f,\min}$ is the minimum mass fraction value and $\Delta M_f^{(k)}$ is the mass fraction change, defined as:

$$\Delta M_f^{(k)} = \min\left(\Delta M_{f,\max}^{(k)},\ \max\left(-\Delta M_{f,\max}^{(k)},\ \lambda_{\Delta Mf}^{(k)} \times \varepsilon_d^{(k)}\right)\right). \tag{4.3}$$

The current mass fraction change modulation factor $\lambda_{\Delta Mf}^{(k)}$ is defined as:

$$\lambda_{\Delta Mf}^{(k)} = \max\left(\lambda_{\min},\ \left|\frac{\Delta M_f^{(k-1)}}{\varepsilon_d^{(k-1)} - \varepsilon_d^{(k)}}\right|\right). \tag{4.4}$$

Where $\lambda_{\min}$ is the minimum value for this modulation factor. Notice that $\lambda_{\Delta Mf}^{(k)}$ measures the speed of correction between the mass fraction and the design constraint and modulates the mass fraction change accordingly. Eventually, $\Delta M_{f,\max}^{(k)}$ is a monotonically decreasing function of the iteration k and of modulation factors η and τ. This improves convergence of the algorithm as the permissible change of the mass fraction decreases with the iteration number. $\Delta M_{f,\max}^{(k)}$ can be defined as either a linear function of k:

$$\Delta M_{f,\max}^{(k)} = \left(1 - \frac{k}{k_{\max}+1}\right) \times \eta \times M_f^{(k-1)}, \tag{4.5}$$

or an exponential function of k:

$$\Delta M_{f,\max}^{(k)} = \max\left(\Delta M_l, \eta \times \exp\left(\frac{-k}{\tau}\right) \times M_f^{(k-1)}\right). \tag{4.6}$$

Here, $k_{\max}$ is the maximum iteration number and ΔM_l is the minimum value for $\Delta M_{f,\max}^{(k)}$. These two parameters, as well as the parameters η and τ, need to be defined by the user of the algorithm.

4.3.2 Setpoint update

The IED setpoint is updated within the "inner loop" (*cf.* Figure 4.12) until the mass constraint defined for the next iteration is respected. For each step j of the "inner loop", the setpoint $S^{*(j,k)}$ is given by:

$$S^{*(j,k)} = \min\left(S^*_{\max}, \max\left(S^*_{\min}, S^{*(j-1,k)} \times \frac{m^{(j-1)}}{M_f^{(k)}}\right)\right). \tag{4.7}$$

Where $m^{(j-1)}$ is the mass fraction derived from the thickness updates of step $j-1$ and $S^*_{\min}$ and $S^*_{\max}$ are the IED setpoint limits which prevent the setpoint from reaching incredibly high or low values.

4.3.3 Neighbourhood

Given the initial list of cells, the neighbourhood of each cell is initialised as follows: search for existing in-plane neighbours, then, search for existing out-of-plane neighbours (see for instance Figure 4.11(b) and Figure 4.11(d)).
The boundary cells have an incomplete neighbourhood (their neighbours do not exist outside the design space). In some HCA implementations (as described by [53]), the lattice is virtually extended to complete the neighbourhoods of the boundary cells. As such virtual extensions do not have a physical meaning, they are not used in HCATWS.
Notice that, during the optimisation, when a cell's thickness is under the minimum value permissible $t_{\min}$, it is removed from the finite element model and from the neighbourhoods it belongs to. Therefore, it does not have an influence on the update of its neighbours any more.

4.3.4 Cells update rules

Approach of Tajs-Zielińska

This first update rule was introduced in [18], hence it is referred to as the Tajs-Zielińska update rule. It is detailed here as it allows to treat the update contribution for each cell of the neighbourhood separately. This is especially interesting for the HCATWS method and its macro cells. At step j of the "inner loop" and iteration k

of the "outer loop", the thickness of cell i is updated as follows:

$$\delta t_i^{(j,k)} = \max\left(-\delta t_{\max},\ \min\left(\delta t_{\max},\ \sum_{q=0}^{n_i} \alpha_q^{(j,k)}\right)\right). \tag{4.8}$$

Where $\delta t_{\max}$ is the maximum thickness variation allowed per iteration, n_i is the number of neighbours of cell i, $\alpha_0^{(j,k)}$ is the update contribution of the current cell and for $1 \leq q \leq n_i$, $\alpha_q^{(j,k)}$ is the update contribution of the neighbouring cell q. These contributions are defined as:

$$\alpha_0^{(j,k)} = \begin{cases} +C_0 & \text{if} \quad U_0^{(k)} > S^{*(j,k)} \\ -C_0 & \text{if} \quad U_0^{(k)} < S^{*(j,k)} \end{cases}, \qquad \alpha_q^{(j,k)} = \begin{cases} +C_q & \text{if} \quad U_q^{(k)} > S^{*(j,k)} \\ -C_q & \text{if} \quad U_q^{(k)} < S^{*(j,k)} \end{cases}. \tag{4.9}$$

Where $S^{*(j,k)}$ is the current IED setpoint and $U_i^{(k)}$ is the IED of cell i. C_0 and C_q are the update increments for, respectively, the current cell and the neighbouring cells. Hence, the new thickness of cell i is given by:

$$t_i^{(j,k)} = \begin{cases} \max\left(t_{\min},\ \min\left(t_{\max},\ t_i^{(k-1)} + \delta t_i^{(j,k)}\right)\right) & \text{if} \quad t_{\min} < t_i^{(k-1)} \leq t_{\max}, \\ \max\left(0,\ \min\left(t_{\max},\ t_i^{(k-1)} + \delta t_i^{(j,k)}\right)\right) & \text{if} \quad 0 \leq t_i^{(k-1)} \leq t_{\min}. \end{cases} \tag{4.10}$$

Where $t_{\min}$ and $t_{\max}$ are the user defined boundaries for thickness. For a thickness below $t_{\min}$, the cells are removed from the finite element model.

Hybrid approach

This second rule is the one implemented in the HCATWS method. It is referred to as hybrid rule since it is inspired both by the update rule presented in [18] (see previous section) and the update rule presented in [97]. This rule does not only account for the contributions of the neighbouring cells, but also accounts for the discrepancy between the setpoint and the cells' IED levels. This update rule also follows Equation (4.8) and Equation (4.10), but the update contributions are modified as follows:

$$\begin{aligned} \alpha_0^{(j,k)} = &\quad \sum_{u=1}^{l_0} \zeta \times \mathbf{1}_{P_u}\left(\tilde{U}_0^{(k)}\right) - \sum_{u=1}^{l_0} \zeta \times \mathbf{1}_{M_u}\left(\tilde{U}_0^{(k)}\right). \\ \text{where} \quad P_u = &\quad \left[S^{*(j,k)} + b_{0,u} \times \text{disc}_{\text{up}}^{(j,k)};\ S^{*(j,k)} + \text{disc}_{\text{up}}^{(j,k)}\right]. \\ M_u = &\quad \left[S^{*(j,k)} - \text{disc}_{\text{low}}^{(j,k)};\ S^{*(j,k)} - b_{0,u} \times \text{disc}_{\text{low}}^{(j,k)}\right]. \\ \text{and} \qquad &\forall\, 1 \leq u < l_0, \quad 0 < b_{0,u} < b_{0,u+1} < 1. \end{aligned} \tag{4.11}$$

for the current cell,

$$\begin{aligned} \alpha_q^{(j,k)} = &\quad \sum_{v=1}^{l_q} \zeta \times \mathbf{1}_{P_v}\left(\tilde{U}_q^{(k)}\right) - \sum_{v=1}^{l_q} \zeta \times \mathbf{1}_{M_v}\left(\tilde{U}_q^{(k)}\right). \\ \text{where} \quad P_v = &\quad \left[S^{*(j,k)} + b_{q,v} \times \text{disc}_{\text{up}}^{(j,k)};\, S^{*(j,k)} + \text{disc}_{\text{up}}^{(j,k)}\right]. \\ M_v = &\quad \left[S^{*(j,k)} - \text{disc}_{\text{low}}^{(j,k)};\, S^{*(j,k)} - b_{q,v} \times \text{disc}_{\text{low}}^{(j,k)}\right]. \\ \text{and} \qquad &\quad \forall\, 1 \leq v < l_q, \quad 0 < b_{q,v} < b_{q,v+1} < 1. \end{aligned} \tag{4.12}$$

for each in-plane neighbouring cell q, and

$$\begin{aligned} \alpha_r^{(j,k)} = &\quad \sum_{w=1}^{l_r} \zeta \times \mathbf{1}_{P_w}\left(\tilde{U}_r^{(k)}\right) - \sum_{w=1}^{l_r} \zeta \times \mathbf{1}_{M_w}\left(\tilde{U}_r^{(k)}\right). \\ \text{where} \quad P_w = &\quad \left[S^{*(j,k)} + b_{r,w} \times \text{disc}_{\text{up}}^{(j,k)};\, S^{*(j,k)} + \text{disc}_{\text{up}}^{(j,k)}\right]. \\ M_w = &\quad \left[S^{*(j,k)} - \text{disc}_{\text{low}}^{(j,k)};\, S^{*(j,k)} - b_{r,w} \times \text{disc}_{\text{low}}^{(j,k)}\right]. \\ \text{and} \qquad &\quad \forall\, 1 \leq w < l_r, \quad 0 < b_{r,w} < b_{r,w+1} < 1. \end{aligned} \tag{4.13}$$

for each out-of-plane neighbouring cell r.
In the three previous equations, $\mathbf{1}_A$ is the indicator function of subset A ($\forall x \in A$, $\mathbf{1}_A(x) = 1 \,\text{and}\, \forall x \notin A$, $\mathbf{1}_A = 0$). l_0, l_q and l_r are the number of subdivisions of the IED intervals for respectively the current cell, the in-plane neighbour cells and the out-of-plane neighbour cells. $b_{0,u}$, $b_{q,v}$ and $b_{r,w}$ are the corresponding subdivisions levels. These subdivisions allow to rank the cells depending on their distance to the setpoint. ζ is an increment parameter which modulates the thickness changes depending on the mass fraction change of the current iteration. It is defined as:

$$\zeta = \max\left(\zeta_{\min}, \frac{t_0\, \Delta M_f^{(k)}}{\theta}\right). \tag{4.14}$$

Where $\zeta_{\min}$ is the minimum increment allowed, t_0 is the initial cell thickness value (related to the initial mass fraction) and θ is is an integer parameter ensuring that the amount of thickness changes for a given iteration is correlated to the mass fraction change of this iteration. $\text{disc}_{\text{up}}^{(j,k)}$ and $\text{disc}_{\text{low}}^{(j,k)}$ are the extreme discrepancies between the setpoint and the cells IED levels. They are defined by comparing the setpoint to the n_c cells of the model:

$$\text{disc}_{\text{up}}^{(j,k)} = \max_{1 \leq i \leq n_c}\left(U_i^{(k)}\right) - S^{*(j,k)}, \;\text{ and }\; \text{disc}_{\text{low}}^{(j,k)} = S^{*(j,k)} - \min_{1 \leq i \leq n_c}\left(U_i^{(k)}\right). \tag{4.15}$$

Eventually, to avoid oscillation of the algorithm, the IED of cell i at iteration k,

$U_i^{(k)}$, is replaced by $\tilde{U}_i^{(k)}$, a weighted sum of the three previous iterations:

$$\tilde{U}_i^{(k)} = \frac{1}{2} \times U_i^{(k)} + \frac{1}{3} \times U_i^{(k-1)} + \frac{1}{6} \times U_i^{(k-2)}. \tag{4.16}$$

Empty neighbourhood

For empty neighbourhoods, the update rule is based on the hybrid update rule described above. Given the empty neighbourhood, Equations (4.12) and (4.13) are disregarded.

Reintroduction of removed cells

In some situations, to avoid getting stuck in a local optimum, it can be necessary to enforce the reintroduction of deleted cells. With the non-empty update rules described previously, the reintroduction of cells is possible through the influence of neighbours. Nevertheless, if one wants to reintroduce cells all over the model, this can be enforced as described here. Under the current implementation of the algorithm, the user is able to choose using this option or not. When choosing to do so, the update rules described previously are applied only when the displacement or force constraint is not violated. When it is violated, all the cells under $t_{\min}$ see their thickness increase by the same value $\delta t_{\mathrm{R}}^{(j,k)}$ defined as:

$$\delta t_{\mathrm{R}}^{(j,k)} = \min\left(\delta t_{\max}, \frac{t_0\, S^{*(0)}}{M_f^{(k)}\, S^{*(j,k)}}\right). \tag{4.17}$$

In this equation, $S^{*(0)}$ is the initial IED setpoint defined as:

$$S^{*(0)} = \frac{\mathrm{KE}_0}{n_c\, \mathrm{M}_0}. \tag{4.18}$$

Where KE_0 is the initial kinetic energy of the model and M_0 the initial mass of the design space. This increase in thickness should enforce the reintroduction of some or all the deleted cells. Notice that for the other cells of the model, the normal update rule is applied.
Since the IED levels of the cells are averaged over three iterations (see Equation (4.16)), the risk of this enforced reintroduction leading to an oscillation of the algorithm is lowered.

Symmetries

In some cases, it is advantageous to use a symmetry condition to obtain symmetric structures. When such a condition is used, after the normal update rule is applied, symmetrical cells are updated once more and assigned the same thickness (e.g. if cell n and m are symmetrical, $t'_n = t'_m = 0.5 \times (t_n + t_m)$).

4.4 Implementation

4.4.1 Architecture

The optimisation algorithm is implemented in the numerical programming and computation software Scilab [108]. The user specifies the optimisation parameters in the input file `hcatws_user_call.sce` (see Section A.1), the optimisation run is also started with this file. The crash simulation parameters and results to extract are defined in the input file `compute_and_extract.sce` (see Section A.2). If needed, they can be modified during the optimisation which can be useful if the computation resources change during the optimisation. Eventually, the user can specify which values should be monitored during the optimisation process in the input file `monitoring.sce` (see Section A.3).
Each new iteration of the optimisation process is run in a new folder. After the thickness distribution update by the algorithm, the list of cells' thickness is exported in the file `cells.list` (see Section A.4). Cells with a thickness value under $t_{\min}$ are added to the file `delcell.del` (see Section A.4) for deletion in the model. Using these two lists, a batch commands file is generated for automatic update of the geometry and creation of the finite element model by SFE CONCEPT (*cf.* Section 4.4.2). The simulation is then run with the finite element solver LS-DYNA [80]. The main results are grouped and organised in two different text files for easier reading by Scilab. `global_output.out` gathers the relevant outputs for the whole model (e.g. mass, impactor displacement, impactor reaction force...) while `cells_output.out` gathers the specific outputs relevant to each cell (e.g. IED) (see Section A.5). With the outputs contained in these two files, Scilab can proceed to the next iteration.

4.4.2 Automatic update with SFE CONCEPT

The geometry is generated using the CAE software SFE CONCEPT [109]. This software allows to model parametric geometries and generate finite element models. It can also be run in batch mode which is especially adapted to integrate it within an optimisation environment. Such an integration is presented for instance in [59], where SFE CONCEPT is used within a shape optimisation environment.
Within the topology optimisation environment presented here, SFE CONCEPT allows an automatic update of the geometry (addition or removal of cells, which modifies the topology), update of the cells' thickness, update of the connections between cells and generation of new finite element model. It also enables a quicker transition between the topology optimisation result and an optional additional shape optimisation (see for instance Section 5.5.3).

Naming conventions

To enable the automatic updates of the model, a naming convention is defined. The cubic boxes generating the space filling are identified with their x,y,z position and are each made of a separate component in SFE CONCEPT (C_1_1_1 identifies the box with the lowest x,y,z position).
As mentioned in Section 4.2.2, each of these boxes may have 3, 4, 5 or 6 walls. Let the point M be the origin node of box C_X_Y_Z and l the size of the boxes. The coordinates of M are x_M, y_M, z_M. The point M' is defined with the following coordinates, $x_{M'}$=$x_M + l$, $y_{M'} = y_M + l$, $z_{M'} = z_M + l$. Each cell of box C_X_Y_Z is assigned a specific number which is also used in the finite element model as each cell has a specific Property ID (PID). As shown in Figure 4.13, the cells are coded as follows: the cell in yz plane through point M is assigned number $XYZ00$ (red cell), the cell in xy plane through point M is assigned number $XYZ01$ (orange cell), the cell in xz plane through point M' is assigned number $XYZ02$ (blue cell), the cell in xy plane through point M' is assigned number $XYZ03$ (yellow cell), the cell in xz plane through point M is assigned number $XYZ04$ (green cell) and the cell in yz plane through point M' is assigned number $XYZ05$ (purple cell).

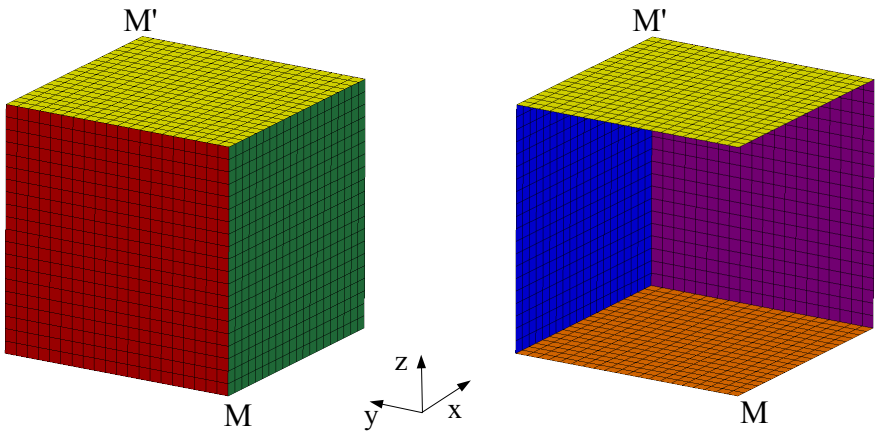


FIG. 4.13 – Naming convention of cells within a box. Cells $XYZ00$, $XYZ03$, $XYZ04$ (*left*) and cells $XYZ01$, $XYZ02$, $XYZ03$, $XYZ05$ (*right*).

Update

To update the model, the fully filled design space is always taken as a starting point and the modifications are done on the geometry. The mesh is then generated and the finite element model is exported. Using the aforementioned naming convention, the cells' thickness can be updated by modifying the thickness of each PID. Relevant cells can be removed from the model by setting the corresponding surface as passive in the geometrical model. Connections between cells can also be updated if necessary.

4.4.3 User parameters

PROBLEM PARAMETERS			
name	type	typical value	definition
n_c	integer	none	Eq. 4.15
KE_0	float	none	Eq. 4.18
$\mathrm{d}_{\max}$	float	none	Eq. 4.1
M_0	float	none	Eq. 4.18
ALGORITHM PARAMETERS			
$t_{\min}$	float	0.4 (mm)	Eq. 4.10
$t_{\max}$	float	3.0 (mm)	Eq. 4.10
$k_{\max}$	integer	100	Eq. 4.5
ε_1	float	0.0001 (J/kg)	Fig. 4.12
ε_2	float	0.1 (mm)	Fig. 4.12
η	float	0.3	Eq. 4.5 or 4.6
τ	integer	50	Eq. 4.6
$M_{f,0}$	float	1.0	Sec. 4.4.3
$M_{f,\min}$	float	0.01	Eq. 4.2
ΔM_l	float	0.01	Eq. 4.6
$\delta t_{\max}$	float	0.5 (mm)	Eq. 4.8
$S^*_{\min}$	float	0.0001 (J/kg)	Eq. 4.7
$S^*_{\max}$	float	$\mathrm{KE}_0/\mathrm{M}_0$ (J/kg)	Eq. 4.7
$\zeta_{\min}$	float	0.01 (mm)	Eq. 4.14
θ	integer	3	Eq. 4.14
l_0	integer	5	Eq. 4.11
l_q	integer	3	Eq. 4.12
l_r	integer	2	Eq. 4.13
$b_{0,u}$, $b_{q,v}$, $b_{r,w}$	float	none	Eq. 4.11 to 4.13

TAB. 4.1 – List of user parameters

The previous table summarises the user parameters of the optimisation method. They are divided into optimisation problem parameters and algorithm parameters. Whenever possible, a typical value of the parameter is given.

Some parameters are more critical than others. The thickness boundaries can be seen either as problem or algorithm parameters. They can be the manufacturing limits of the problem or can be artificial limits used to drive the algorithm to different topologies. $k_{\max}$, ε_1, ε_2, η and τ have a direct influence on the convergence of the algorithm. η and τ should be adjusted to make sure that a satisfying solution can be reached within $k_{\max}$ iterations. The initial mass fraction $M_{f,0}$ is an important user parameter as it defines the starting thickness distribution within the cells. Indeed, the mass fraction is the mass objective divided by the reference structure, which is usually defined as the whole space filling with a thickness of 1.0 mm. $\delta t_{\max}$ is important too as it is the maximum thickness variation allowed per iteration per cell. It can be seen as the potential evolution speed of the algorithm. θ needs to be chosen carefully as it relates ζ and $\Delta M_f^{(k)}$. Its value should be modified with

the neighbourhood type chosen. Indeed, with fewer neighbours it should be lower in order to increase the influence of each cell in the update rule. The update rules parameter l_0, l_q, l_r, $b_{0,u}$, $b_{q,v}$ and $b_{r,w}$ are important as they are at the heart of the algorithm.

$S^*_{\min}$ and $S^*_{\max}$ are not critical parameters. They are only defined to make sure that the setpoint does not reach too high or too low values. Similarly, lower boundaries such as $M_{f,\min}$, ΔM_l, $\zeta_{\min}$ are defined to prevent the computation from getting stuck.

Chapter 5

Optimisation of extruded structures using HCATWS

In this chapter, the HCATWS method presented previously is adapted for the design of extruded structures. Although these structures are still 3D structures, the definition of their topology is a 2D problem and the HCATWS method needs to be adapted accordingly.

Extruded structures are extensively used in automotive applications. They allow to generate complex geometries with only a few manufacturing operations. Unlike stamped metal sheets, they allow to generate closed cross-sections without gluing or welding process.

Two types of extruded structures can be mentioned. Self-contained extruded structures (see Figure 5.1) and reinforcement extruded structures where an extruded pattern is added to an already existing structure in order to reinforce it (see Figure 5.2). Both types of extruded structures are used for energy absorption but the first type is the most documented with many prismatic beam examples available in the journal *Thin Walled Structures*. Such examples include for instance the work of Kim [70], Zhang *et al.* [129] or Hou *et al.* [57] where the energy absorption under axial impact of multi cell extruded beams is characterised.

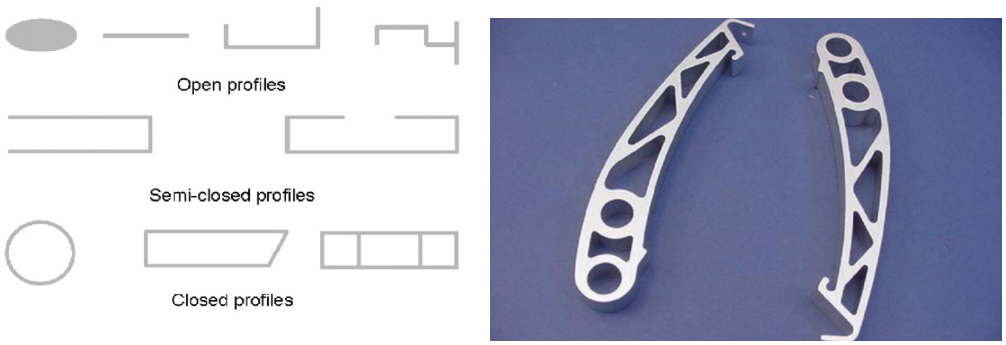


FIG. 5.1 – Extruded structures: basic cross-section types (*left*) and example of a pedal cut from an extrusion (*right*) (*Source: [43]*).

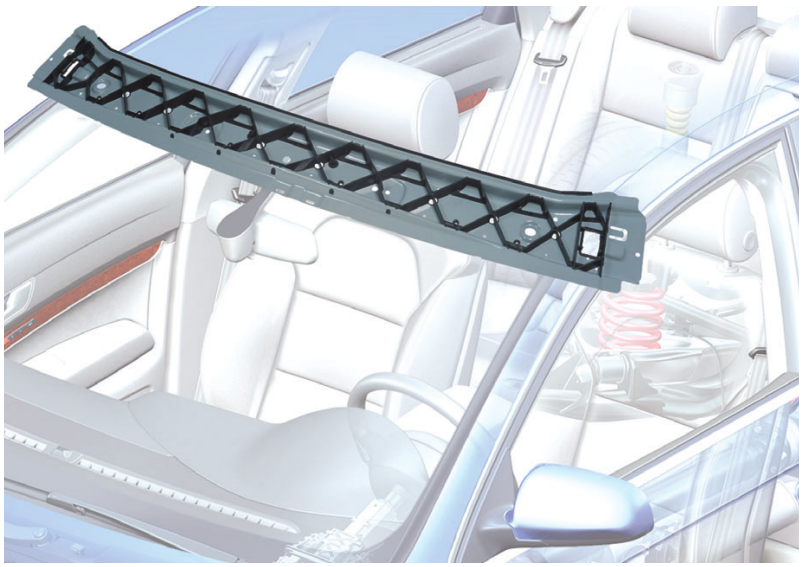


FIG. 5.2 – Example of reinforcement extruded structure: metallic roof beam reinforced with extruded plastic ribs (*Source: [65]*).

In Section 5.1, the modifications necessary to adapt the method presented in Chapter 4 to the design of extruded structures are detailed. Then different application examples are presented from Section 5.2 to Section 5.5. They include a static shear-torsion case and different dynamic crash cases.

5.1 Adapting the method for 2D design spaces

As mentioned in the introduction, in order to use the HCATWS method to design extruded structures, a few modifications should be implemented. While the algorithm itself is not modified, the initial space filling and the related neighbourhood definitions need to be adapted to 2D problems.

5.1.1 Space filling

The problem of space filling with thin walls in 2D is equivalent to defining a tiling or a prismatic honeycomb for an extruded structure. While there is only one regular space filling in 3D, there are three different regular tilings in 2D, the triangular tiling, the square tiling and the hexagonal tiling (see [31] and Figure 5.3). A quasiregular tiling also exists with the triangular-hexagonal tiling (see Figure 5.4). This tiling is edge-transitive (or isotoxal). As the edges of the tiling are the thin walls of the structure, this tiling would generate a regular grid of cells. Another isotoxal tiling to mention is the rhombille tiling (see Figure 5.4). For the HCATWS application, the hexagonal and rhombille tilings are dismissed as they do not allow to generate straight lines. The square tiling is chosen for practicality reasons as it is simply a transfer of the space filling chosen in Chapter 4 for 3D problems. In the future though, the triangular and triangular-hexagonal could be investigated too.

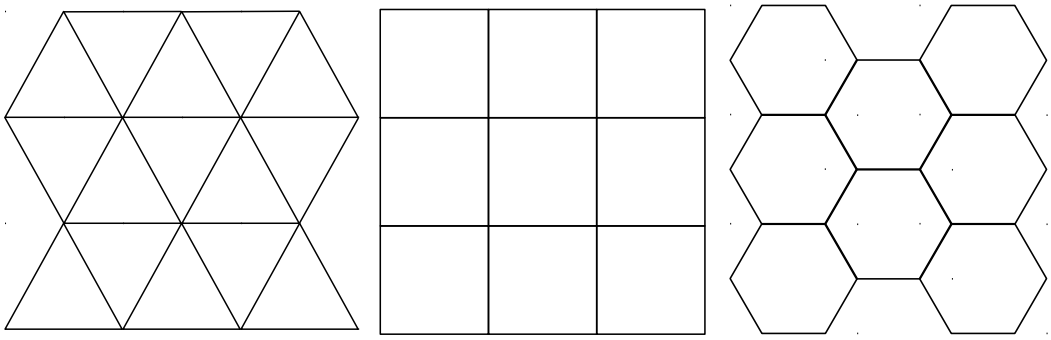


FIG. 5.3 – Regular 2D space fillings: triangular tiling (*left*), square tiling (*centre*) and hexagonal tiling (*right*).

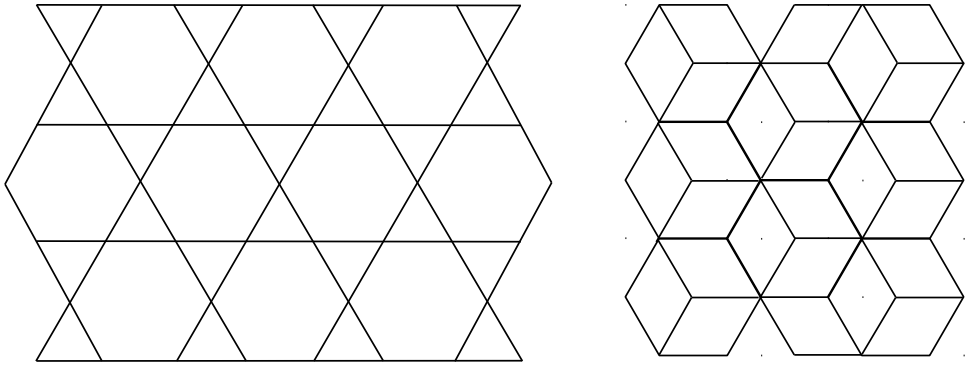
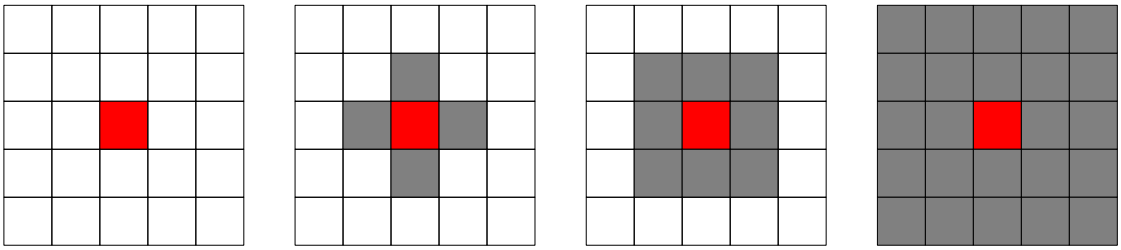


FIG. 5.4 – Alternative 2D space fillings: triangular-hexagonal tiling (*left*) and rhombille tiling (*right*).

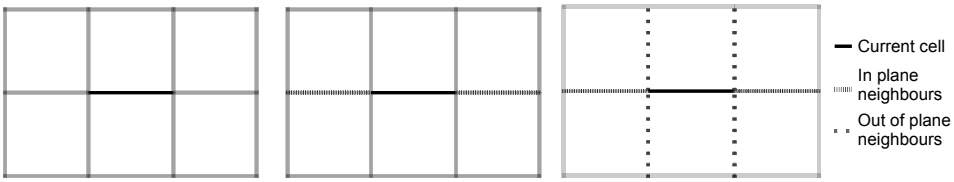
5.1.2 Neighbourhood



(a) Empty (b) Von Neumann (c) Moore (d) Extended Moore

FIG. 5.5 – Classic 2D neighbourhoods (*inspired from [18]*).

The neighbourhoods are defined in a similar manner as described in Section 4.2.3 for 3D problems. They are illustrated in Figure 5.6. For comparison, the classic neigbourhoods for 2D solids are illustrated in Figure 5.5. The *von Neumann* neighbourhood (see Figure 5.6(c)) still contains the cells sharing an edge (the main connection feature for thin walls) with the current cell. When representing only the cross-section of the extrusion though, it looks like the neighbours share only a node.



(a) Empty (b) Plane (c) Von Neumann

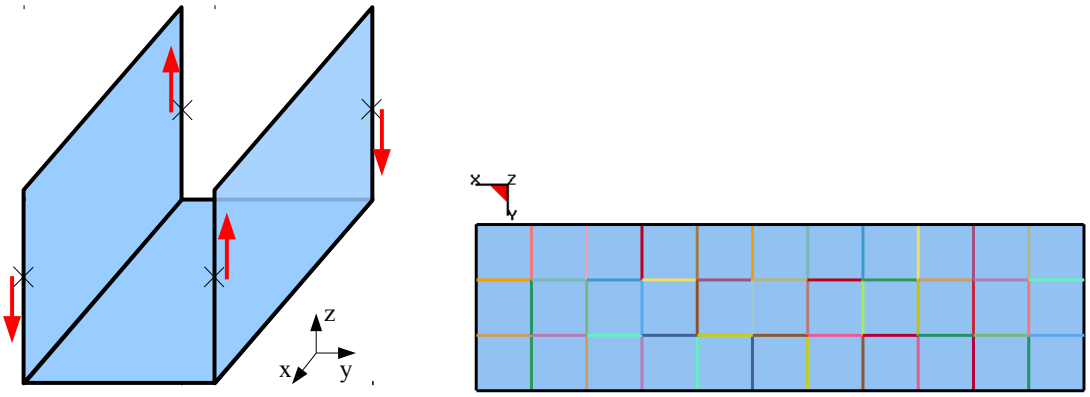
FIG. 5.6 – Neighbourhoods for extruded thin-walled structures.

To differentiate the in-plane neighbour cells from the out-of-plane neighbour cells, a *plane* neighbourhood is introduced (see Figure 5.6(b)) which contains only the cells in the same plane as the current cell.
In terms of implementation, the same script as for the 3D case is used to initialise the neighbourhood of each cell. The neighbourhoods only contain less cells with possibly 2 in-plane neighbours and possibly 4 out-of-plane neighbours for the *von Neumann* neighbourhood.

5.2 First example: shear-torsion static case

5.2.1 Problem definition

This problem is a simple example designed to test different configurations of the algorithm. It is a static load case applied to a U-type cross-section reinforced with some ribs. It can be regarded as a combined load case (see Figure 5.7(a)) which generates both a torsion of the cross-section and shear of the outer panels (see Figure 5.8).
Although the HCATWS method is developed and presented in this thesis as specific to crashworthiness design, it is based on the classic HCA (see Section 3.5) which has been successfully applied for topology optimisation with static load cases (see for instance [18, 118, 100]). It is therefore natural to test the HCATWS method on a static problem too.
The objective of the problem is to optimise the extruded plastic reinforcement pattern within a metallic beam under the shear-torsion load. A comparable approach can be used to derive the structure presented in Figure 5.2. In Figure 5.7(b), the initial space filling for the reinforcements is represented. Each cell is assigned its own properties in the model and is therefore represented here with its own color. As mentioned in Section 5.1 it uses a square tiling.
The dimensions of the metallic beam are $90 \times 90 \times 330$ mm. The forces applied are 100 N per node in z or $-z$-direction. The ends of the beams are free to warp. The material properties (Young modulus, Poisson's ratio and mass density) of the metallic beam are: $E = 210$ GPa, $\nu = 0.3$, $\rho = 8.34$ t/m^3. The material properties of



(a) Load case (b) Square space filling

FIG. 5.7 – Shear-torsion problem definition.

the plastic reinforcement are: $E = 2.3$ GPa, $\nu = 0.35$, $\rho = 1.45$ t/m^3. The objective is to minimise the mass of the reinforcements while keeping the average displacement of the nodes where the load is applied under d_{max} (absolute displacement in z-direction).

For the empty metallic beam, The average displacement for these nodes is 7.2 mm, in order to double the shear-torsion stiffness of the beam, d_{max} is chosen as 3.6 mm. The simulations are run with LS-DYNA. Unless stated, a mesh size of 5 mm is used in the automatic mesher of SFE CONCEPT. The study of such open thin-walled cross-section under torsion is largely treated in literature, see for instance [19] or [8] where the dimensions of given cross-sections are optimised to satisfy a twist angle. The deformation and stress profiles of the empty cross-section are given in Figure 5.9 and Figure 5.10 profile.

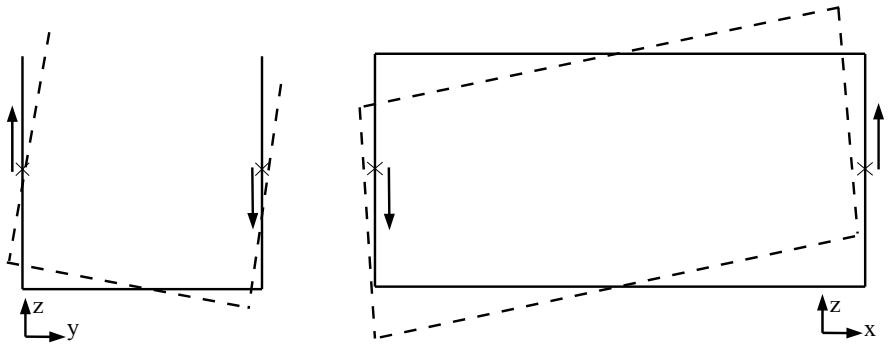


FIG. 5.8 – Deformation directions induced by the load case, cross-section view (*left*) and side view (*right*).

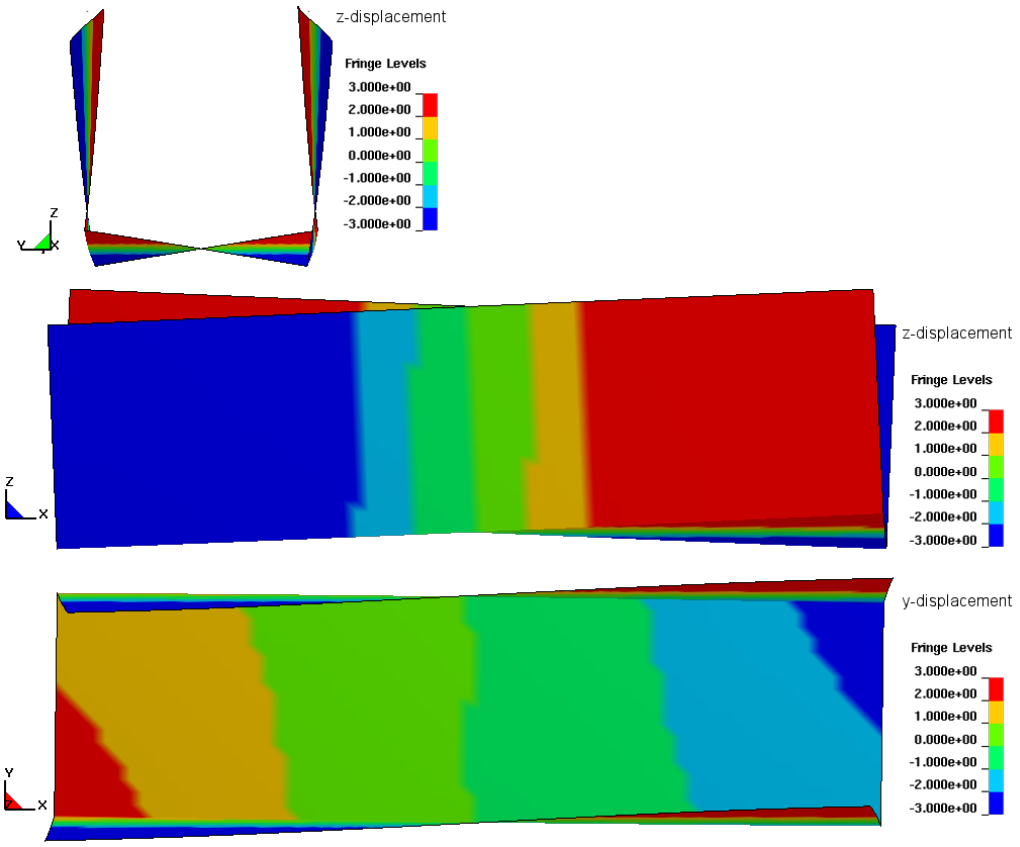


FIG. 5.9 – Deformation profile of the empty cross-section (node displacements in mm).

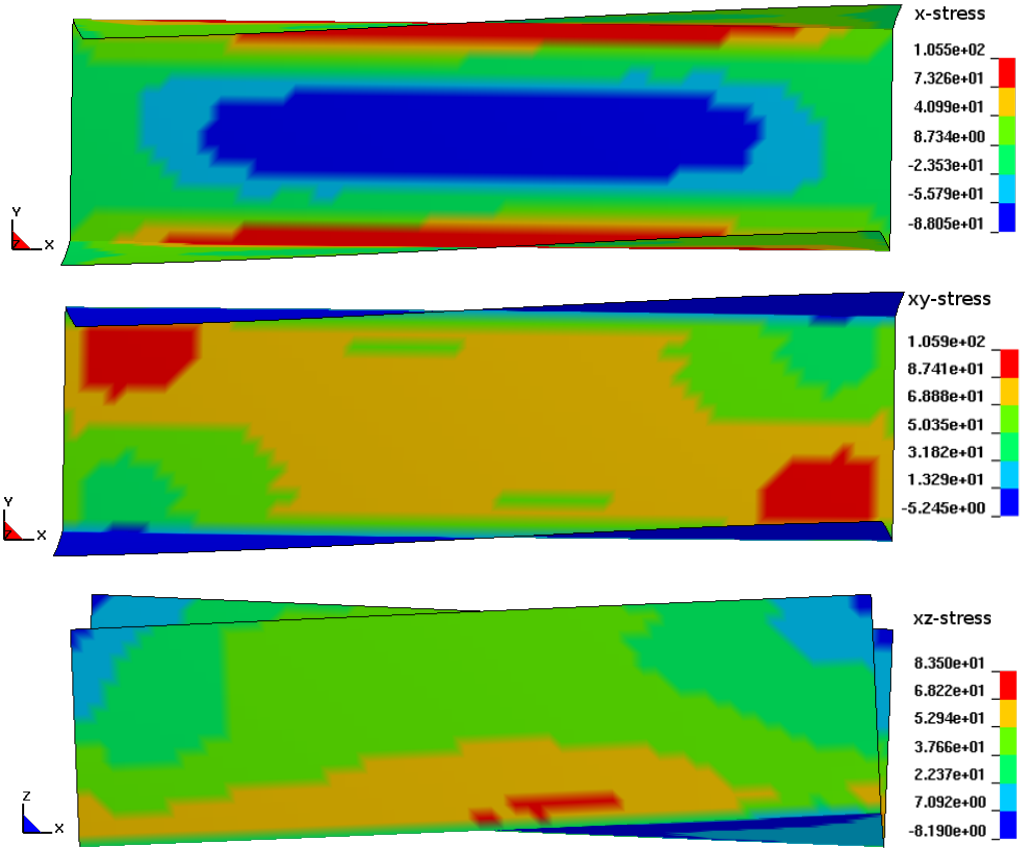


FIG. 5.10 – Main stress profiles of the empty cross-section (stresses in MPa).

5.2.2 Optimisation with square space filling

The space filling illustrated in Figure 5.7(b) uses 3 cells (*i.e.* walls) in y-direction and 11 cells in x-direction for a total of 52 cells (there are no walls at the ends of the beam). Given the dimensions of the design space, each cell has a width of 30 mm and an extrusion length of 90 mm (z is the extrusion direction). Given the symmetries of the load, a central symmetry constraint is applied to the update rule of the cells (see Section 4.3.4).

Empty neighbourhood

To begin with, the problem is solved using an empty neighbourhood. This setting is tested for further comparison with the non-empty neighbourhood. As mentioned in Section 4.3.4, a reintroduction mechanism is used here for deleted cells. This avoids getting stuck in a local minimum. This mechanism is particularly important with the empty neighbourhood as when a cell is deleted, there is no influence from its neighbours to promote its reintroduction when necessary.
The starting thickness for this optimisation run is 1.0 mm for all cells ($M_{f,0} = 1.0$). The maximum mass fraction change allowed $\Delta M_{f,\max}$ is set constant equal to 0.15. The cells' thickness boundaries are set to $t_{\min} = 0.4$ mm and $t_{\max} = 3.0$ mm and the maximum thickness variation per iteration is set as $\delta t_{\max} = 0.5$ mm. The hybrid update rule is used (see Section 4.3.4). The update coefficients in Equation (4.11) are set as: ζ is constant equal to 0.1 mm, $b_{0,1} = 0.05$, $b_{0,2} = 0.1$, $b_{0,3} = 0.2$, $b_{0,4} = 0.4$, $b_{0,5} = 0.7$. The parameters are summarised in Table 5.1.

$t_{\min}$	$t_{\max}$	$k_{\max}$	ε_1	ε_2	$M_{f,0}$	$\delta t_{\max}$	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$	$b_{0,5}$
0.4	3.0	200	1e-4	0.1	1.0	0.5	0.05	0.1	0.2	0.4	0.7

TAB. 5.1 – Parameters for first run.

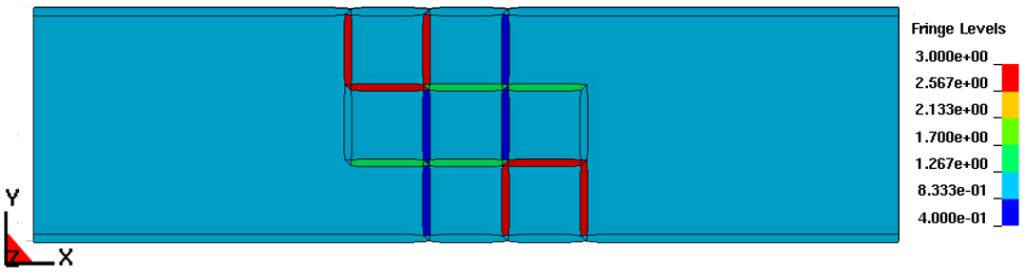


FIG. 5.11 – Best design for first optimisation run, thickness distribution (mm).

For this first run, the best design is found at iteration 40 (see Figure 5.11) with a mass of 850 g, and mass increase of 14.4% (with respect to the non-reinforced beam). The optimisation is stopped manually at iteration 72 after getting stuck in a local minimum of a higher mass than for iteration 40 (see Figure 5.12). The

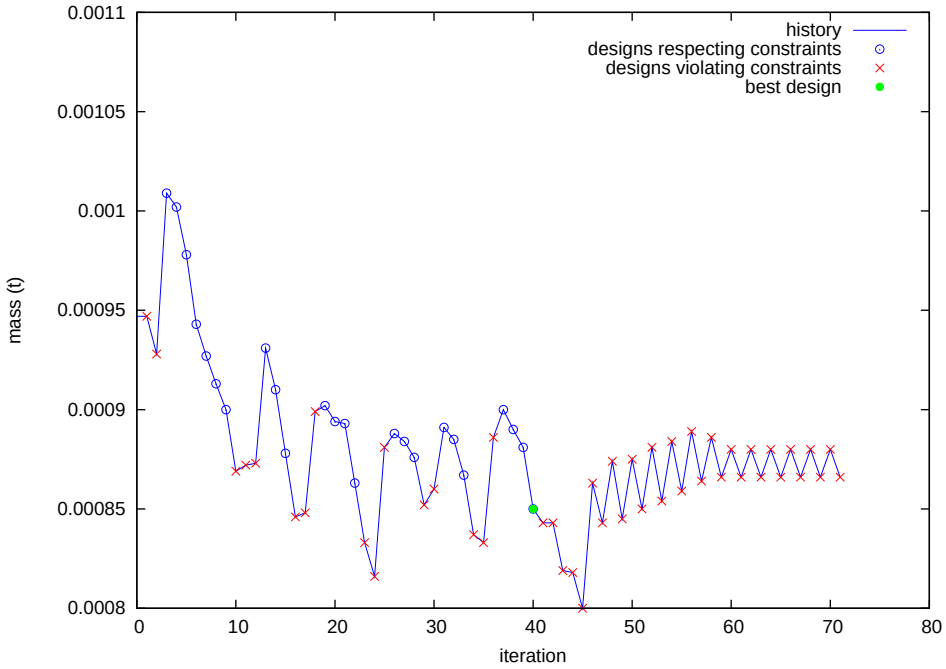


FIG. 5.12 – Optimisation history, first run, mass value.

reinforcements for the best design are located around the middle of the structure where the deformation directions induced by the load case change orientations (see Figure 5.8). Such a reinforcement seems adequate to control the global deformation of the structure.

In a second optimisation run, the definition of the maximum mass fraction change allowed is modified. Instead of being constant as in the first optimisation it is defined as a decreasing function of iteration k (see Equation (4.5)), where $\eta = 0.21$. Also ζ is not constant any more but defined as in Equation (4.14), where $\theta = 3$. The best design is found at iteration 136 (see Figure 5.13) with a mass of 876 g, and a mass increase of 17.9%.

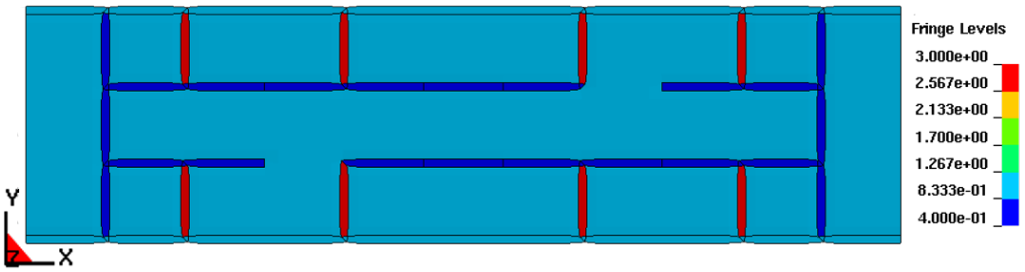


FIG. 5.13 – Best design for second optimisation run, thickness distribution (mm).

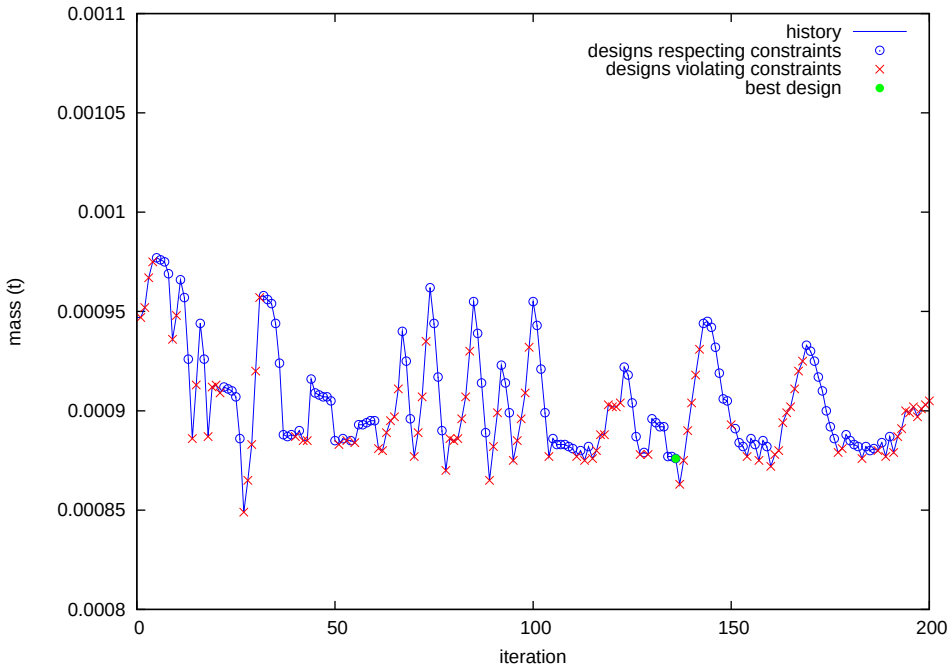


FIG. 5.14 – Optimisation history, second run, mass value.

The mass value is higher than in the first optimisation run and the topology of the best design is also more complex. Figure 5.14 illustrates the mass history throughout the optimisation. The reduction in maximum mass fraction change in the end of the optimisation is clearly visible as expected with the new definition of $\Delta M_{f,\max}$. Figure 5.15 and Figure 5.16 gather the deformation profiles of the best design of the first two runs. They can be compared to the deformation profile of the empty cross-section (see Figure 5.9). For the second run's best design, as the reinforcements are away from the middle of the structure, they tend to follow the deformation of the outer section. For the first run's best design though, the reinforcements connect both sides of the structure going in opposite directions and therefore prevent violating the design constraint.

Von Neumann neighbourhood

After the empty neighbourhood, the von Neumann neighbourhood is investigated. In a third optimisation run, the maximum mass fraction change allowed $\Delta M_{f,\max}$ is set constant equal to 0.15 as in the first run. ζ is also constant equal to 0.1 mm. The update rule coefficients in Equation (4.12) and Equation (4.13) are chosen as $l_q = 2$, $b_{q,1} = 0.1$ and $b_{q,2} = 0.5$ for the in-plane neighbours. For the out-of-plane neighbours $l_r = 1$ and $b_{r,1} = 0.1$. With this setting, the influence of the in-plane neighbours on the update rule is higher than the influence of the out-of-plane neighbours. Indeed, the in-plane neighbours can contribute to the update of a cell by two increments ζ while the out-of-plane neighbours can only contribute by one incre-

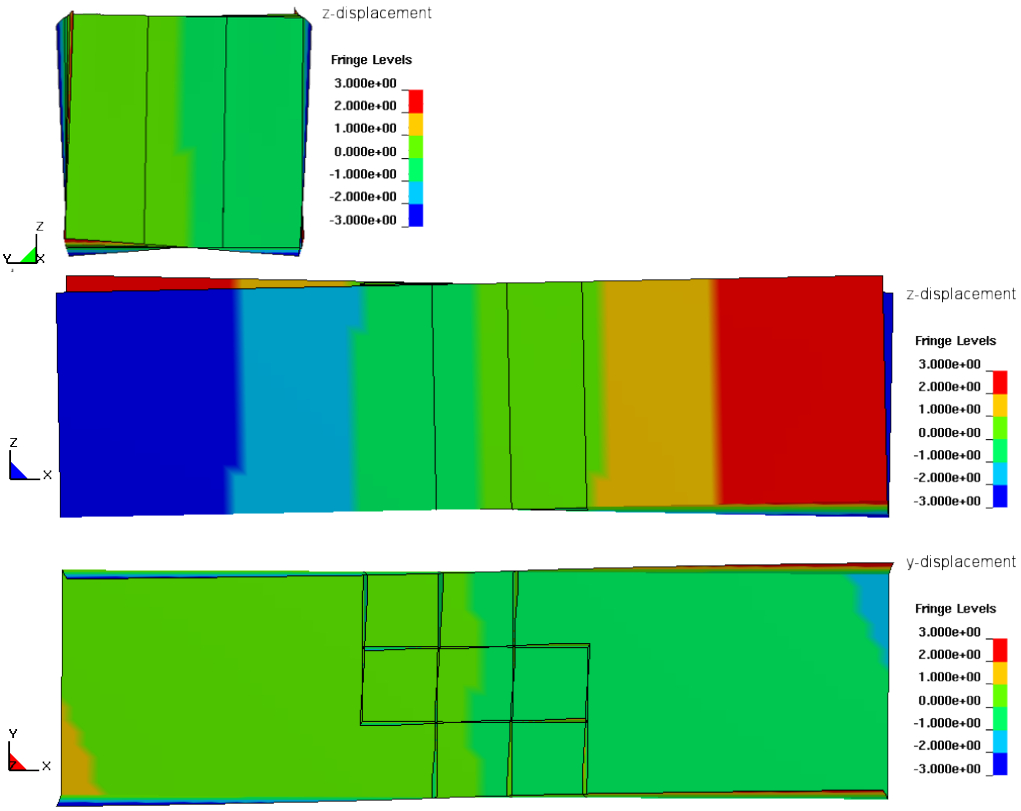


FIG. 5.15 – Deformation profile of the first run's best design (node displacements in mm).

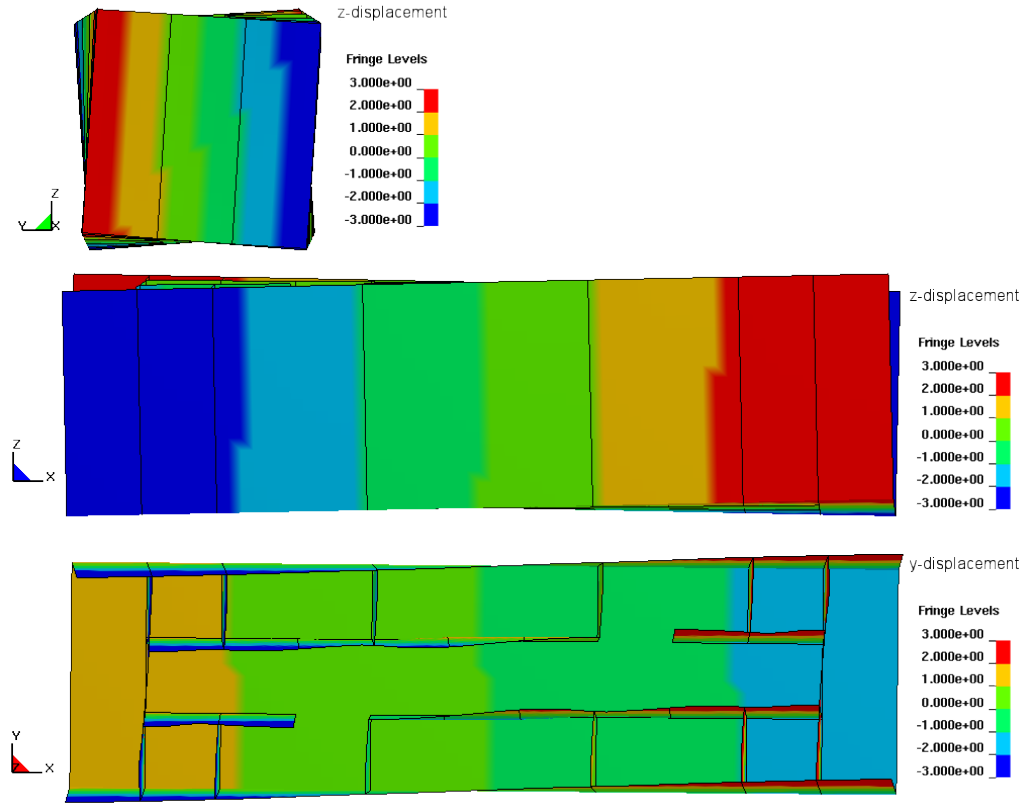


FIG. 5.16 – Deformation profile of the second run's best design (node displacements in mm).

ment ζ. This setting is chosen to favour the generation of plane structures by the algorithm.

The best design for this third optimisation run is found at iteration 67 (see Figure 5.17 and Figure 5.18) with a mass of 830 g or a mass increase of only 11.7%. In a fourth optimisation run, the maximum mass fraction change and the increment ζ are defined as in the second run. The best design for this fourth optimisation run is found at iteration 178 (see Figure 5.19 and Figure 5.20) with a mass of 846 g or a mass increase of 13.9%.

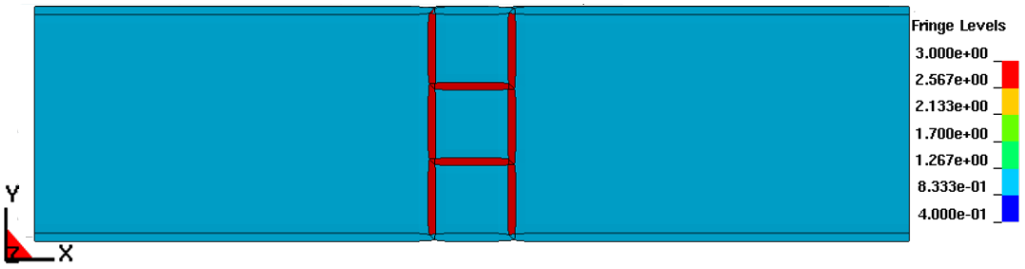


FIG. 5.17 – Best design for third optimisation run, thickness distribution (mm).

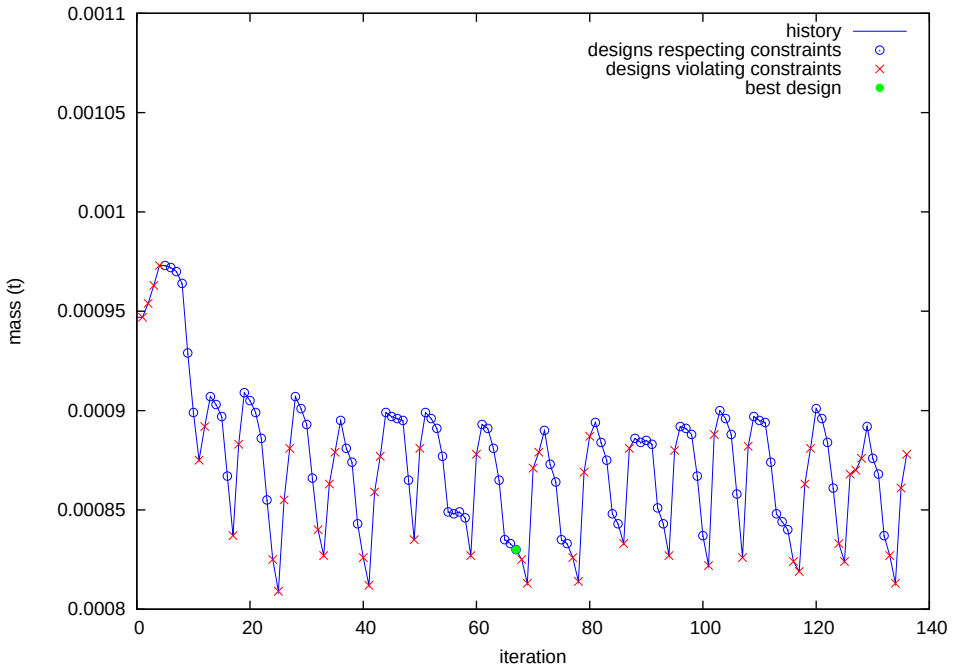


FIG. 5.18 – Optimisation history, third run, mass value.

Table 5.2 gathers the best design results for the four previous optimisation runs. It would be wrong to judge these tests only by the performance of their best design, hence this table should be read along with the optimisation histories and topologies derived from each run. It seems that the non empty neighbourhood helps reintroducing cells more efficiently. Indeed, when a design violates the displacement constraint, it takes less iterations to get back to a design respecting the displacement

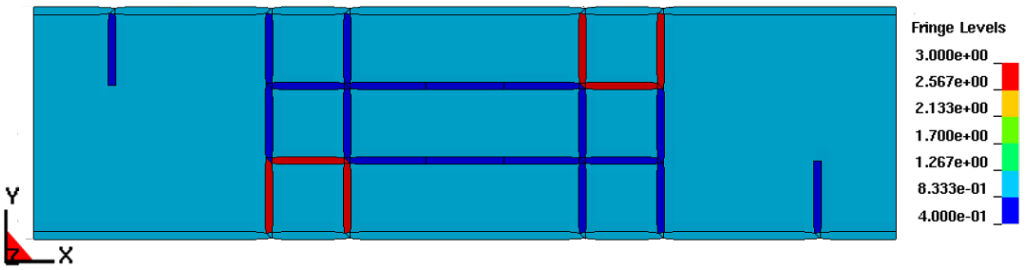


FIG. 5.19 – Best design for fourth optimisation run, thickness distribution (mm).

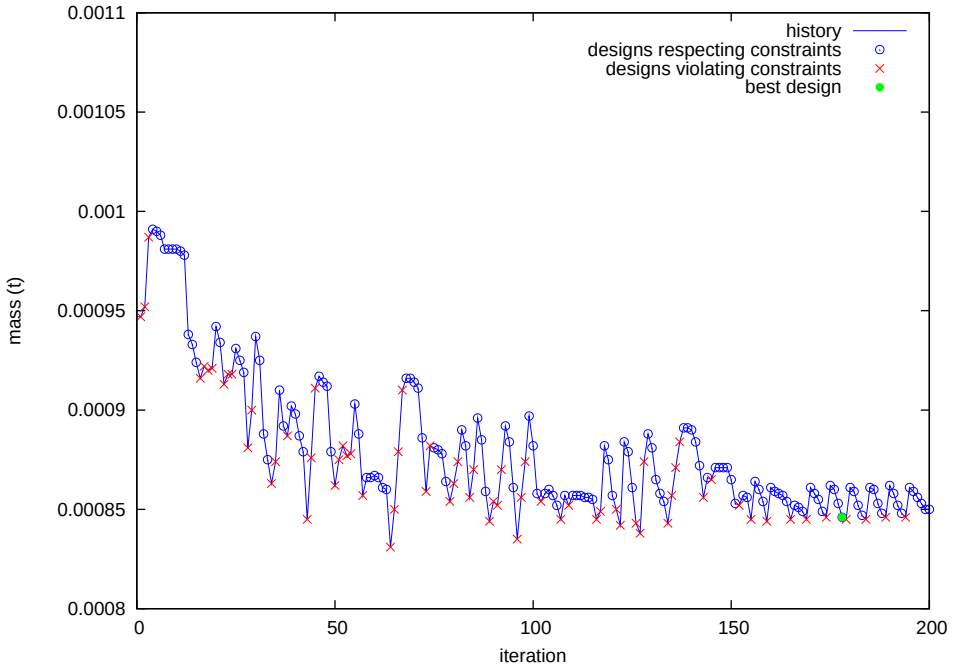


FIG. 5.20 – Optimisation history, fourth run, mass value.

constraint. Also, the definition of $\Delta M_{f,\max}$ as a decreasing function of iteration k favours convergence of the algorithm.

Even though the best design found in the fourth run is not the best in terms of mass, the use of the von Neumann neighbourhood along with the decreasing maximum mass fraction allowed are kept as a base for the following studies.

Run	Design number	Mass (g)	Mass increase (%)	Neighbourhood
1	40	850	14.4	empty
2	136	876	17.9	empty
3	67	830	11.7	von Neumann
4	178	846	13.9	von Neumann

TAB. 5.2 – Best designs for optimisation runs 1 to 4.

Influence of starting thickness

Using the settings of the fourth optimisation run, the influence of the starting thickness on the results is studied. The optimisation is run with six different starting thickness between 0.5 mm and 1.75 mm. The results are summarised in Table 5.3. According to the results, the starting thickness does have a significant influence on the best design. Also, it seems that in this case, the lower the starting thickness is, the better the results are. This could be explained by the fact that in this case, a low starting thickness gives an initial mass fraction close to the optimum mass fraction. The optimisation history of the best run (starting thickness of 0.5 mm) is given in Figure 5.21. The evolution of the topology is also represented. Between iteration 14 and iteration 26, the reintroduction is occurring at iterations 16, 17, 18, 23 and 24. As a result, there are more active cells in iteration 26 than in iteration 14. Also, from iteration 26, the best topology could already be guessed.

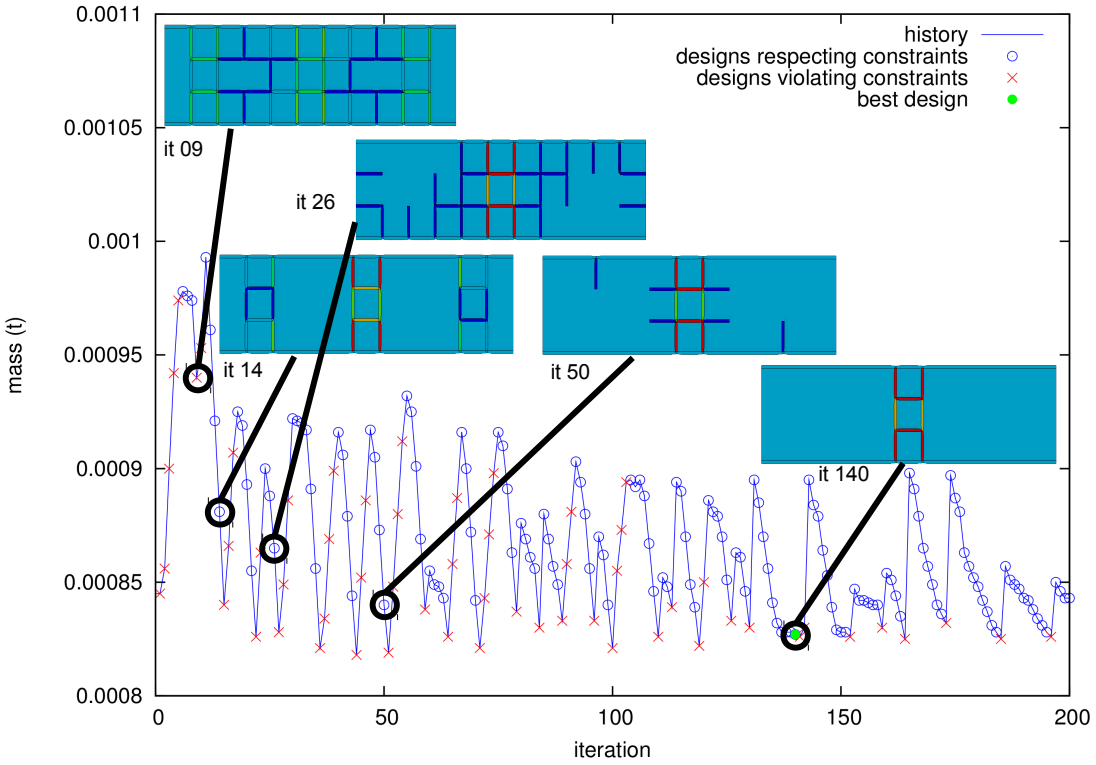


FIG. 5.21 – Optimisation history, best run, mass value and topology evolution.

$M_{f,0}$	Best design	Mass (g)	Mass increase (%)
0.50	140	827	11.3
0.75	192	828	11.4
1.00	178	846	13.9
1.25	163	833	12.1
1.50	91	857	15.3
1.75	169	858	15.5

TAB. 5.3 – Influence of starting thickness on best design.

Influence of cell size

After testing the influence of the starting thickness, the influence of the cell size is investigated. A new space filling is generated with cells twice smaller (with a width of 15 mm instead of 30 mm) than the previous cells. As a result, the new model has 236 cells instead of 52 previously (see Figure 5.22).
The optimisation is run with the same settings as in the fourth run detailed previously, first with $M_{f,0} = 1.0$ and then with $M_{f,0} = 0.5$. The best design is found, in the first run, at iteration 200 with a mass of 834 g or a mass increase of 12.2% (see Figure 5.23) and, in the second run, at iteration 114 with a mass of 822 g or a mass increase of 10.6% (see Figure 5.24). This second result is lighter than the best design found with larger cells. Looking at Figure 5.24, the topology is similar to the one found with larger cells, only with finer discretisation.

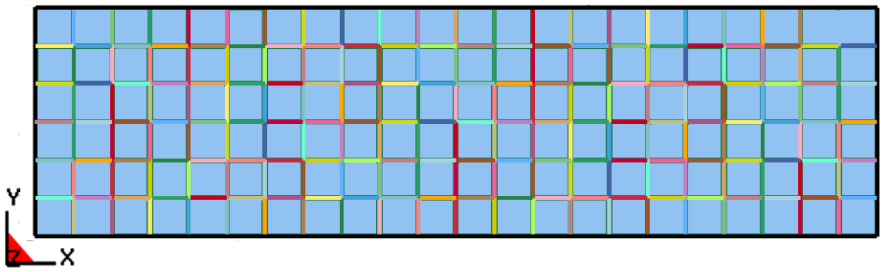


FIG. 5.22 – Space filling with smaller cells.

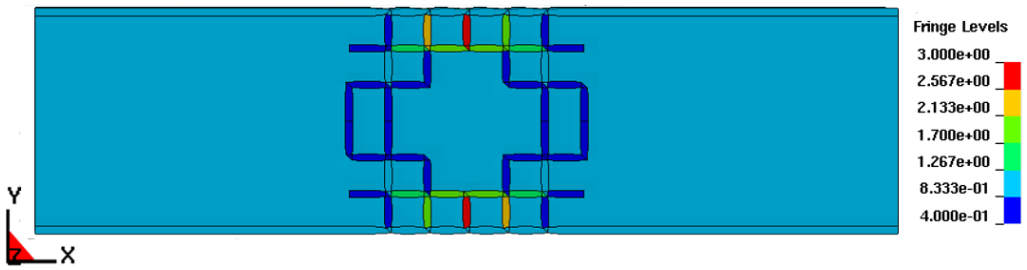


FIG. 5.23 – Best design with smaller cells, thickness distribution (mm).

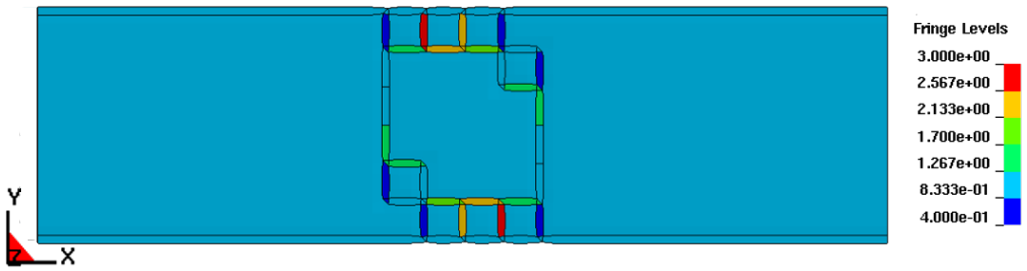


FIG. 5.24 – Best design with smaller cells and lower starting thickness, thickness distribution (mm).

As a comparison, another cell distribution is manually defined where only the cells on the outer part of the beam's cross section remain (see Figure 5.25). This configuration is inspired by the theory of torsion for open thin-walled beams with constant cross-sections. For this C-Channel section, the most efficient way to increase the torsional stiffness would be to increase the thickness of the channel. As this is not a possibility in this design problem, adding cells on the outer part of the cross-section appears to be a good option. The thickness of the cells is set to 0.47 mm in order for the total mass of the beam to be 822 g like the previous best design. In this configuration, the maximum displacement of the nodes is 5.85 mm instead of 3.6 mm. With a trial and error approach, the maximum displacement is then brought down to 3.6 mm increasing the cells thickness to 1.00 mm. In this case, the mass of the beam is 911 g, which is 10.8% more than the best design found with the HCATWS method. This showcases the interest of the method which allows to find a solution with a non constant cross-section, outside of classic torsion theory.

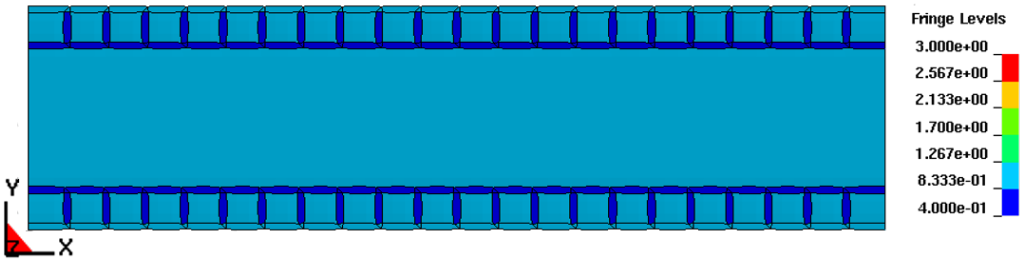


FIG. 5.25 – Alternative cells distribution inspired by theory of torsion, thickness distribution (mm).

Different displacement constraint

To assess the influence of the displacement constraint, the objective is now to triple the shear-torsion stiffness when compared to the empty metallic beam. The displacement constraint is therefore set to $d_{max} = 2.4$ mm. The algorithm settings are identical to the fourth optimisation run described previously. The best design is found at iteration 134 with a mass of 901 g or a mass increase of 21.3 %. As illustrated in Figure 5.26, the topology is similar to the best results found previously with the higher displacement constraint. The structure is again reinforced in the centre to control the torsion of the metallic beam. This time though, more material is needed to satisfy the more demanding constraint.

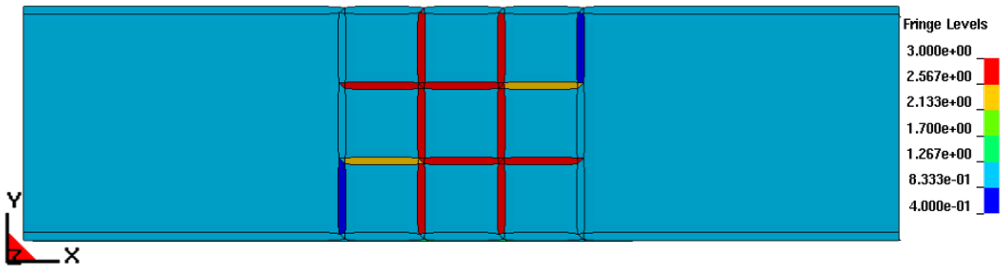


FIG. 5.26 – Best design with lower displacement constraint, thickness distribution (mm).

5.2.3 Rotation of the space filling

The results presented previously suggest that diagonal reinforcements could be more efficient to reinforce the metallic beam for this load case. To test this idea, the square space filling is rotated by 45° (see Figure 5.27). The width of the cells is also increased to 31.8 mm. The space filling is now made of 56 cells.

The displacement constraint is set back to $d_{max} = 3.6$ mm and optimisation settings are almost identical to the fourth optimisation run described previously. The maximum mass fraction change allowed differs as it is now a decreasing exponential function of the iteration number (see Equation (4.6)). The parameters of this function are: $\eta = 0.3$ and $\tau = 50$. Also, the reintroduction of deleted cells is not enforced when the displacement constraint is violated.

The best design is found at iteration 35, with a mass of 791 g or a mass increase of 7.8% (see Figure 5.28). In a second optimisation run for this rotated space filling, the initial mass fraction $M_{f,0}$ is set to 0.5 and the reintroduction of cells is this time enforced. The best design is then found at iteration 148, with a mass of 770 g or a mass increase of 3.6% (see Figure 5.29).

In both cases, the mass increase is much lower than with the original space filling. This supports the idea that diagonal reinforcements can be more efficient than square ones for this load case. It should be noticed that the structure is still reinforced in its centre.

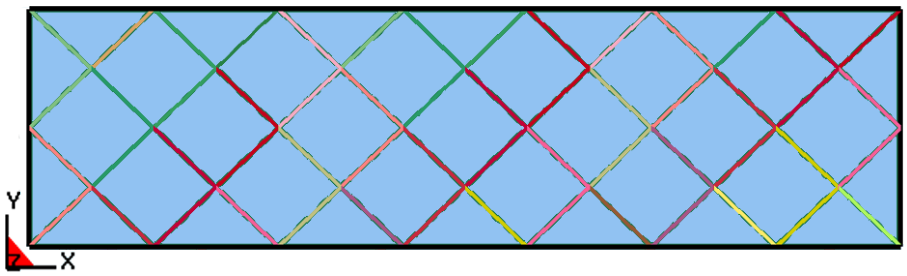


FIG. 5.27 – Rotated square space filling.

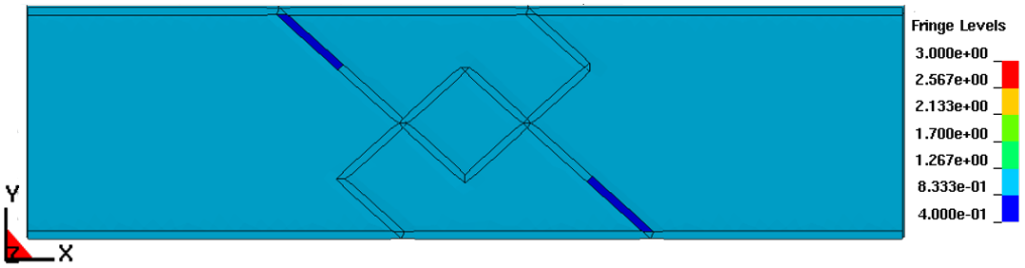


FIG. 5.28 – Best design with rotated space filling (no reintroduction), thickness distribution (mm).

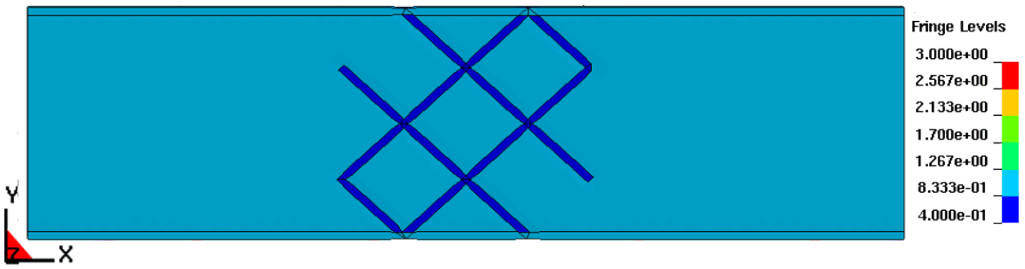


FIG. 5.29 – Best design with rotated space filling (with reintroduction), thickness distribution (mm).

5.3 Second example: axial impact

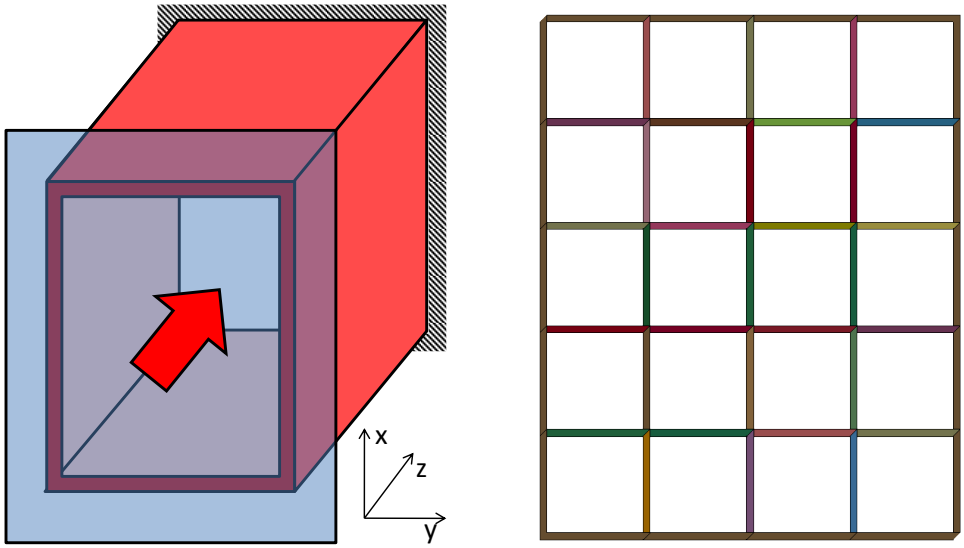
5.3.1 Problem definition

This problem intends to apply the HCATWS to the optimisation of a simplified car component under a transient crash case, as opposed to the static case defined in Section 5.2. The objective is to optimise the cross-section of a prismatic beam for an axial impact load case. As mentioned in the introduction of this chapter, this type of study is well documented in literature. For instance, in the work presented by Kim [70] or by Alavi Nia *et al.* [4], the energy absorption efficiency of different cross-section variants is studied.

The design space is defined by the inner part of a rectangular beam. It has a cross-section of 80×100 mm and a length of 400 mm. The envelope of the beam is fixed with a thickness of 1.5 mm. The design space is filled with a square tiling (see Figure 5.30(b)). The cells have a width of 20 mm and a length of 400 mm which generates a total of 31 cells.

The beam is impacted on one end by a rigid wall perpendicular to its longitudinal direction (see Figure 5.30(a)), and fixed on the other end. The rigid wall has a mass of 500 kg and an initial speed of 5 m/s. In order to keep a reasonable computation time, the impact is simulated for a duration of 25 ms. The displacement of the impactor over this duration is monitored. In real applications case, the peak force between the impactor and the beam should be monitored too. To ensure an axial

folding of the beam during the crash, a symmetry condition is applied to the cells, both on the Oyz and Oxz planes. As a result, the number of independent cells is actually 10.
The beam is made of an aluminium extrusion. The material characteristics are $E = 70$ GPa, $\rho = 2.7$ t/m^3, $\nu = 0.33$ and $\sigma_y = 180$ MPa, where σ_y is the yield stress. Beyond the yield stress, the plastic behaviour is defined with a piecewise linear stress-strain curve (see Table 5.4).



(a) Load case (b) Space filling

FIG. 5.30 – Axial impact, problem definition.

$\varepsilon^p_{\text{eff}}$	0	0.01	0.02	0.05	0.10	0.15	0.2	0.4
σ_y (MPa)	180	190	197	211.5	225.8	233.6	238.5	248.5

TAB. 5.4 – Piecewise linear plasticity table for impacted beam, effective plastic strain $\varepsilon^p_{\text{eff}}$ and corresponding yield stress values σ_y.

In order to define the design constraint, the behaviour of two typical cross-sections under this load case is investigated. The first cross-section is the empty cross-section. Its mass is 583 g and the final displacement in z-direction is 111.0 mm. The deformation is illustrated in Figure 5.31, with two characteristic folds on the larger side of the beam and two and a half folds on the shorter side. The initiation of the folds is illustrated in Figure 5.32 with the folding area over the yield strength value.
The second cross-section reinforces the beam with a cross pattern (see Figure 5.33) of 1.0 mm thickness. Its mass is 778 g which is 33.4 % more than the empty cross-section. The final displacement in z-direction is 97.4 mm. With the cross pattern, the beam's cross-section is compartmented into smaller rectangles which induces a shorter characteristics folding length. With a smaller final displacement, the deformation now exhibits three characteristic folds on both sides of the beam. Again, the

initiation of the folds is illustrated (see Figure 5.34 with the folding area over the yield strength value.

For optimisation, the displacement constraint is defined as d_{max} = 75 mm. That is a reduction of 32 % when compared to the empty cross-section and 24 % when compared to the cross shaped reinforcement.

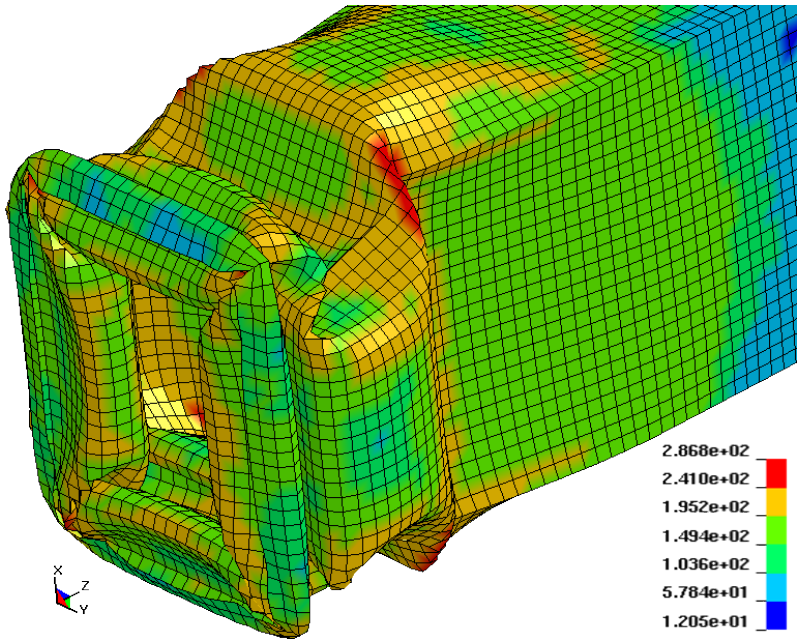


FIG. 5.31 – Empty cross-section, final deformation, von Mises stress (MPa).

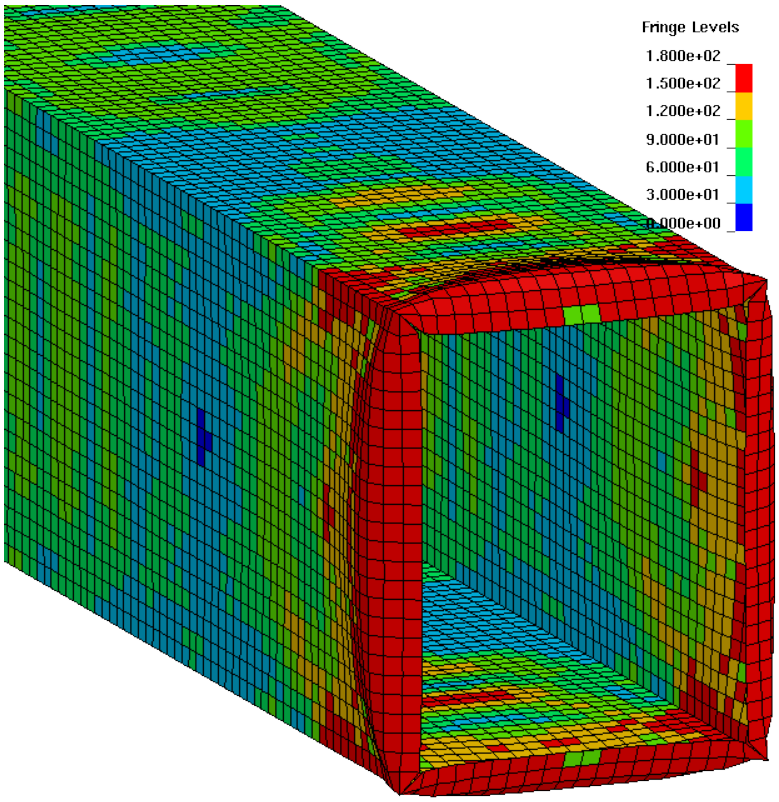


FIG. 5.32 – Empty cross-section, folding initiation, von Mises stress (MPa).

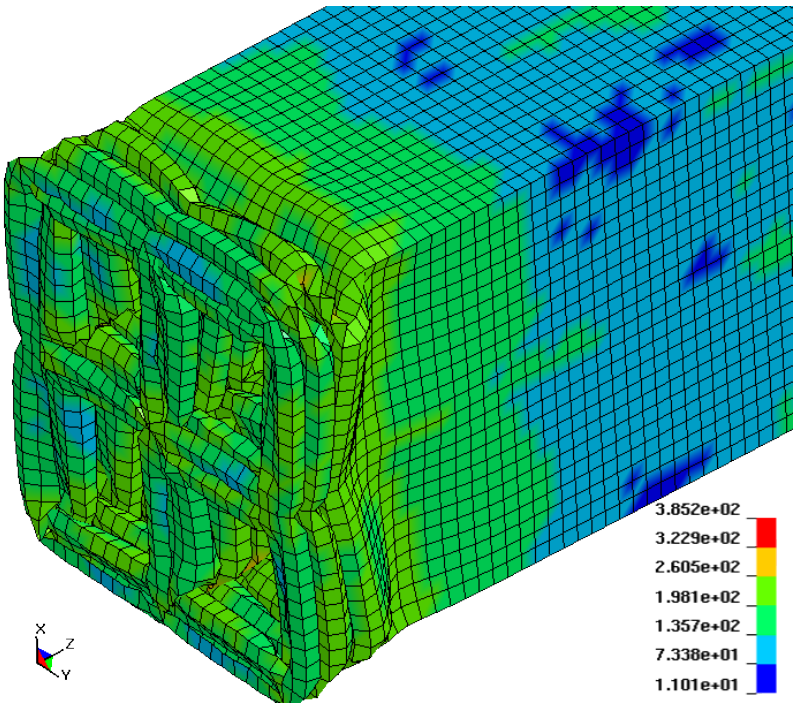


FIG. 5.33 – Cross shaped reinforcement, final deformation, von Mises stress (MPa).

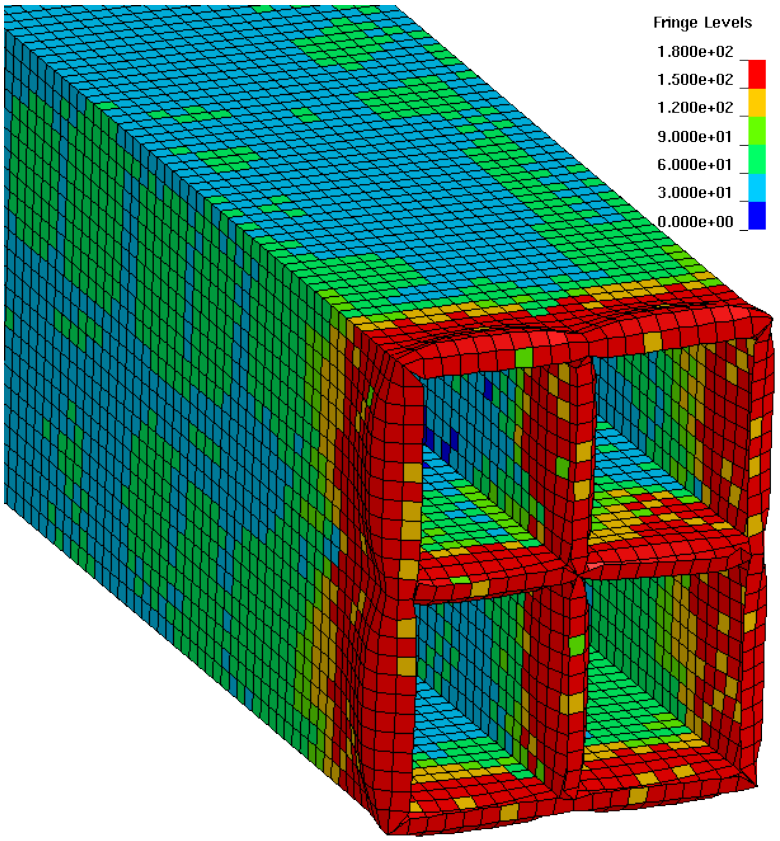


FIG. 5.34 – Cross shaped reinforcement, folding initiation, von Mises stress (MPa).

5.3.2 Topology optimisation results

The topology optimisation is performed with the same settings as the fourth optimisation run detailed in Section 5.2.2. The von Neumann neighbourhood is chosen and the hybrid update rule defined in Section 4.3.4 is used with the same settings as for the shear-torsion load case. Two starting mass fractions are tested (1.0 and 0.5). The best design is found using a starting thickness of 0.5 mm. In that case, the best design is obtained at the 84^{th} iteration with a mass of 878 g or a mass increase of 33.4 % (when compared to the empty cross-section) and a maximum displacement of 74.67 mm (see Figure 5.35).

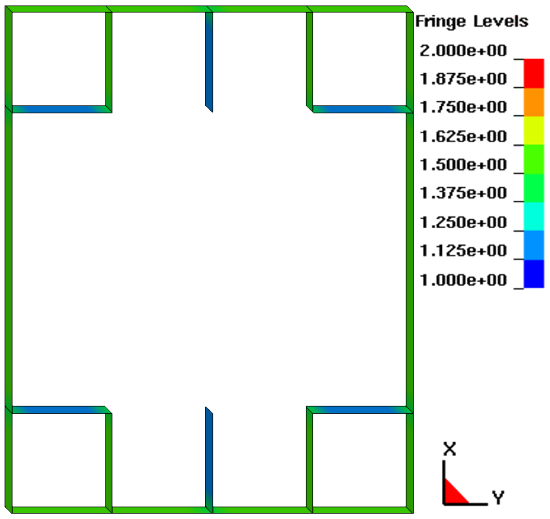
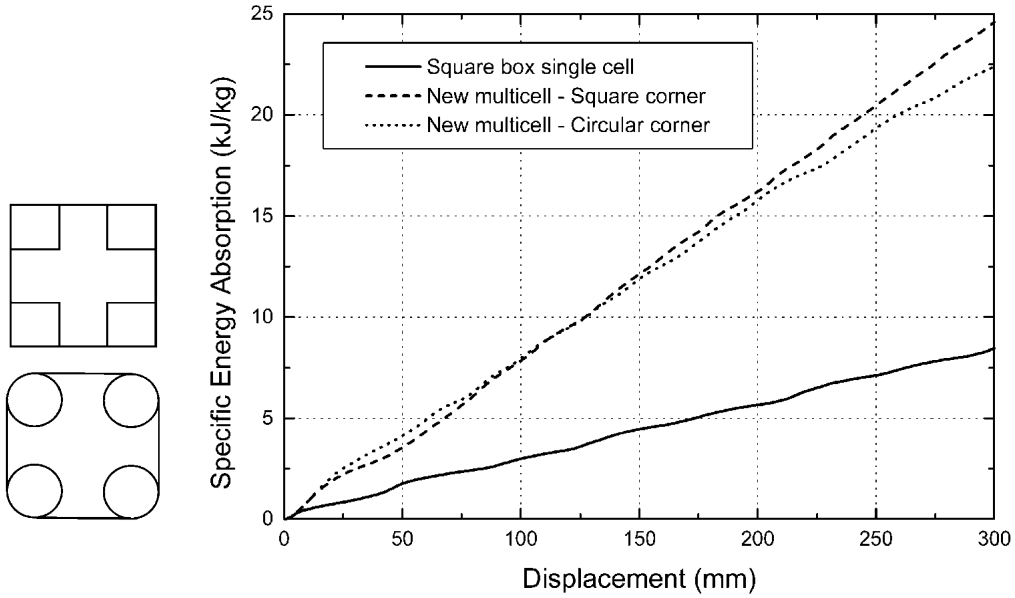


FIG. 5.35 – Optimum design topology (cross-section), thickness distribution (mm).

The topology derived here is interesting as it forms four little square compartments in the corners of the cross-section and some flanges loosely connected to the middle of the shorter beam's sides. Although the beam studied here is not square but rectangular, this result is in agreement with the works of Kim [70] or Alavi Nia *et al.* [4] suggesting that the most efficient way to reinforce a square cross-section for axial energy absorption is to reinforce the corners. In [70] for instance, a 190% increase in Specific Energy Absorption is observed (numerical simulation) when reinforcing a square section with inner square corners (see Figure 5.36).
The final deformation of the beam derived with the HCATWS method is given in Figure 5.37. It is interesting to notice the different number of characteristic folds in different areas of the cross-section. With the shortest segments, four folds are formed in the corners. On the longer side of the beams two folds only are formed. On the shorter side of the beam, with the influence of the loose flanges, the folds are not clearly formed. Also, the initiation of the folds (see Figure 5.38) seems more critical than for the reference cross-sections with the yield strength reached in significant areas along the main edges of the cross-section. For comparison, force-displacement

curves are given in Figure 5.39 for the two reference cross-section and the best design. All cross-sections exhibit a characteristic force displacement curve with a peak force to initialise the folding and then a lower force level to propagate the folding. As expected, the force level is higher for the best design in order to allow for a lower maximum displacement. The energy absorption curves are also given in Figure 5.40.



(a) Reinforced multi-cell cross-sections

(b) Specific Energy Absorption curves

FIG. 5.36 – Axial impact study of multi-cell sections with reinforced corners, comparison with square cross-section (*Source: [70]*).

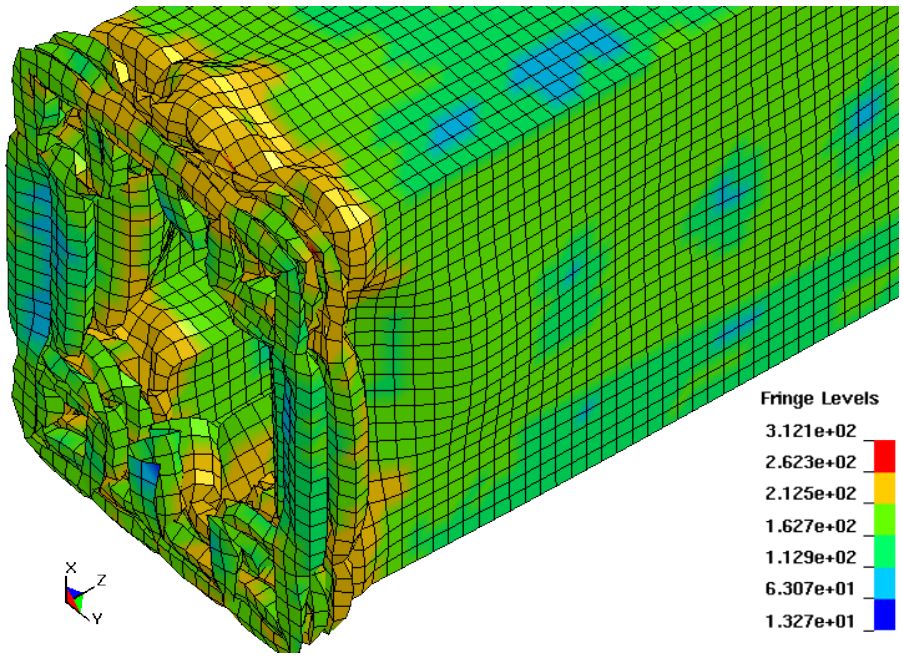


FIG. 5.37 – Optimum design, final deformation, von Mises stress (MPa).

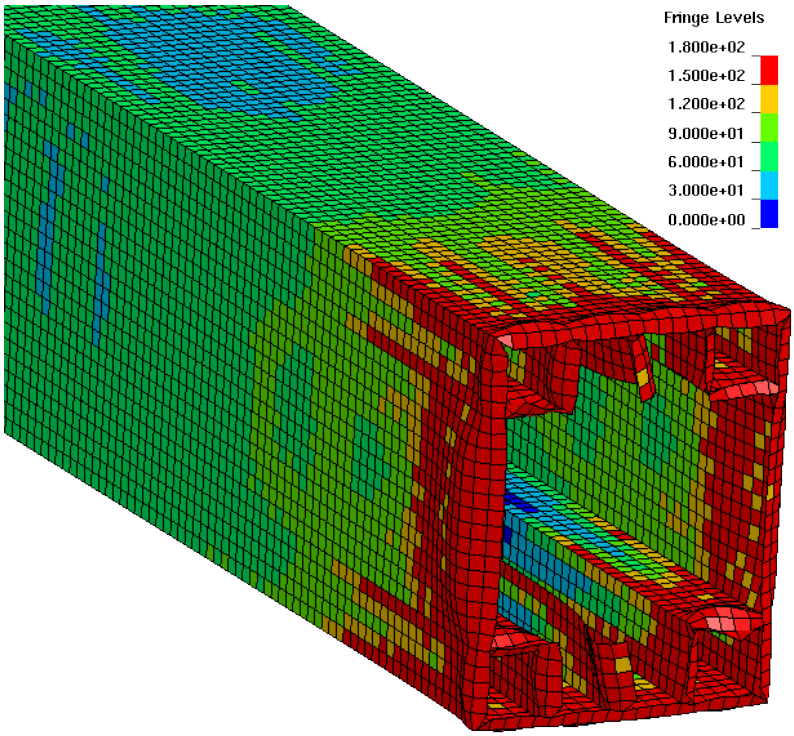


FIG. 5.38 – Optimum design, folding initiation, von Mises stress (MPa).

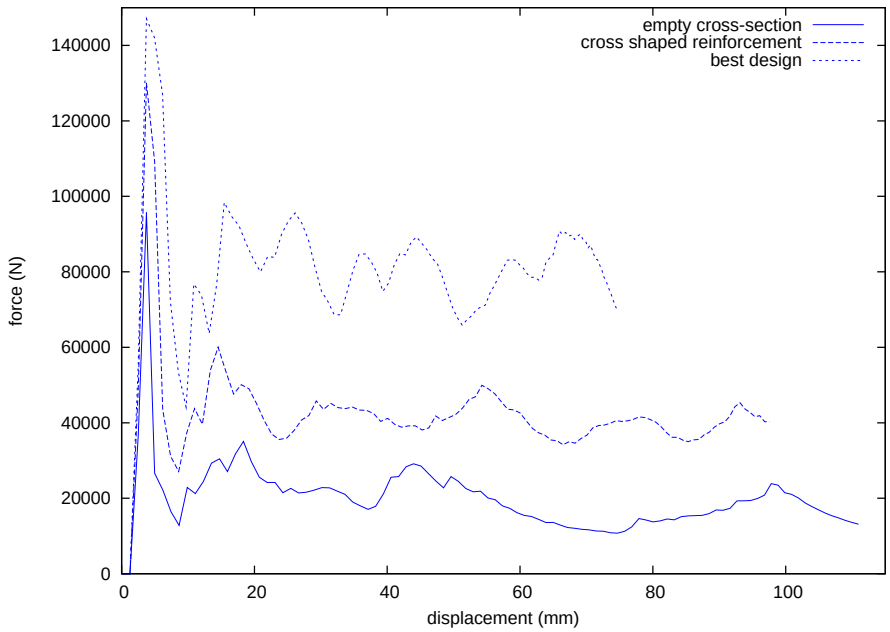


FIG. 5.39 – Force-displacement curves of reference cross-sections and best design.

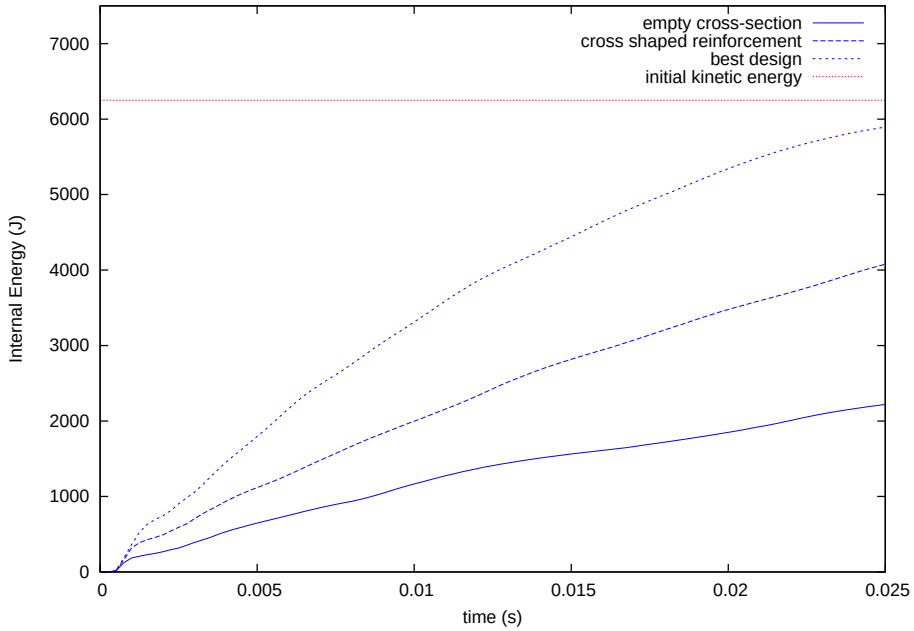


FIG. 5.40 – Energy absorption curves for the reference cross-sections and the best design.

5.3.3 Comparison with enhanced sizing optimisation

As the number of independent cells in this problem is pretty low, it can also be treated as a sizing optimisation problem where the design variables are the cells' thickness and the cells are deleted if the thickness passes the threshold $t_{\min}$. To solve this sizing optimisation problem, the commercial optimisation software LS-OPT [82] is used. The optimisation method chosen is the iterative response surface method implemented in this software.

The fact that the cells can be deleted during the optimisation is not standard for a sizing optimisation. For the response surfaces, this deletion creates a jump in output functions which is usually not desirable. For the sake of comparison though, the problem is solved with this issue in mind. To enable the deletion of the cells, the lower boundary for the thickness parameters is defined as t_{virt}, where $0 < t_{\text{virt}} < t_{\min}$. Practically, t_{virt} is chosen as 0.1 mm in this problem.

As mentioned in the previous section, the optimisation problem presents 10 independent variables and 21 dependent variables. The sampling is made using the D-optimal space filling method (generating 17 points per iteration) and the response surfaces are built using Radial Basis Functions. As for the HCATWS method, the objective is to minimise the mass while keeping the displacement in z-direction under $d_{\max} = 75$ mm. The maximum number of iterations is set to 15 to keep the total number of crash simulations under 300.

The best design was found at iteration 11 with a mass of 863 g (or a mass increase of

48 %) and a maximum displacement of 74.7 mm (see Figure 5.41). The optimisation history is given in Figure 5.42 and the progression of the algorithm can be observed. Although the mass value of this design is better than the one found with the HCATWS method, the topology is similar with reinforcements of the cross-section's corners once again. This is encouraging as in this case, both the theory and the classic optimisation method are in agreement with the result of the HCATWS method. Eventually, the deformation of the beam is presented in Figure 5.43. As for the result of the HCATWS method, four characteristic folds are formed in the corners and two on the longer side of the beam. This time, with the deletion of the flanges on the shorter side of the beam, two and a half folds are clearly formed.

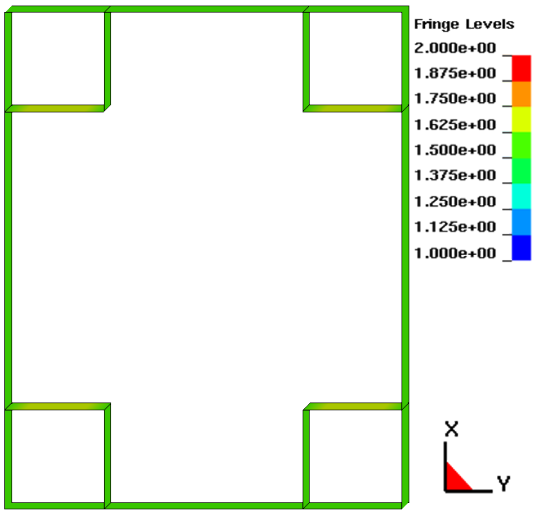


FIG. 5.41 – Enhanced sizing optimisation, best design, thickness distribution (mm).

5.3.4 Different displacement constraint

As for the static shear-torsion problem, a different constraint value is also investigated. In this new case d_{max} is set to 50 mm which is 55% less than for the empty cross-section and 49% less than in the cross shaped reinforcement case.
The optimisation is run with the same settings as previously and once again, two starting thickness are tested, 0.5 mm and 1.0 mm. The best design is found for $M_{f,0} = 1.0$ mm with a mass of 1.125 kg (mass increase of 93.0%) and a maximum displacement of 49.9 mm. The topology (see Figure 5.44) is similar to the topology derived previously with a higher displacement, but this time with thicker cells. The characteristic folds are clearly formed (see Figure 5.45) with two and a half folds along the small compartments and one and a half fold along the longer edge.

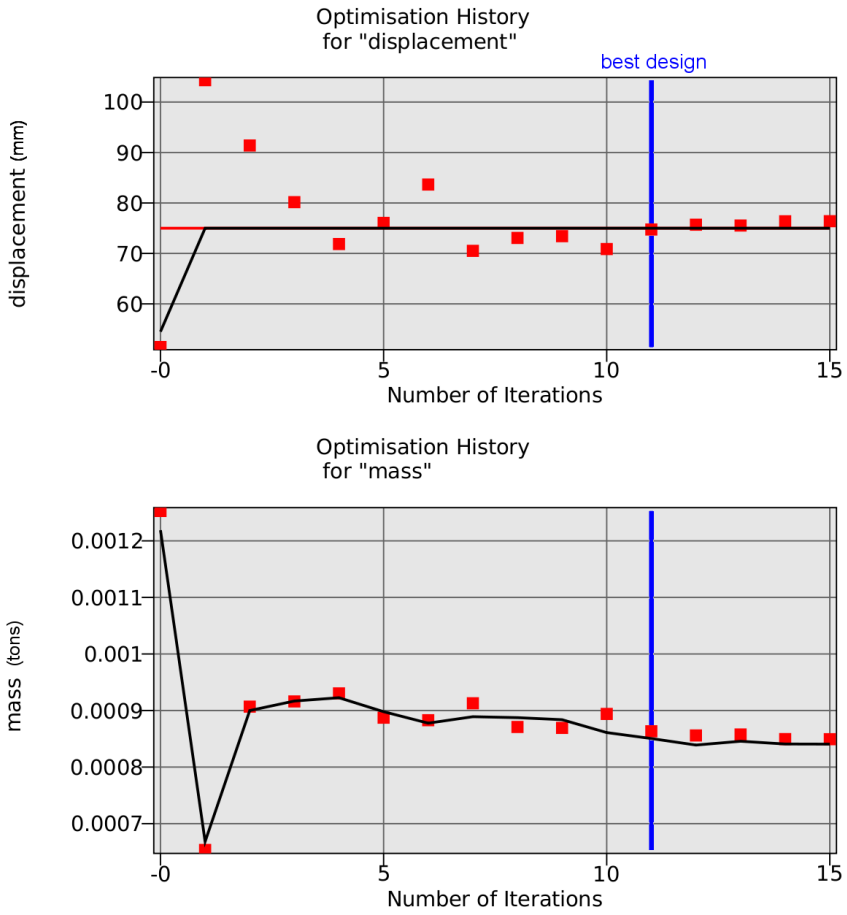


FIG. 5.42 – Displacement (*top*) and mass (*bottom*) history from successive response surfaces (black lines) and validation points (red dots).

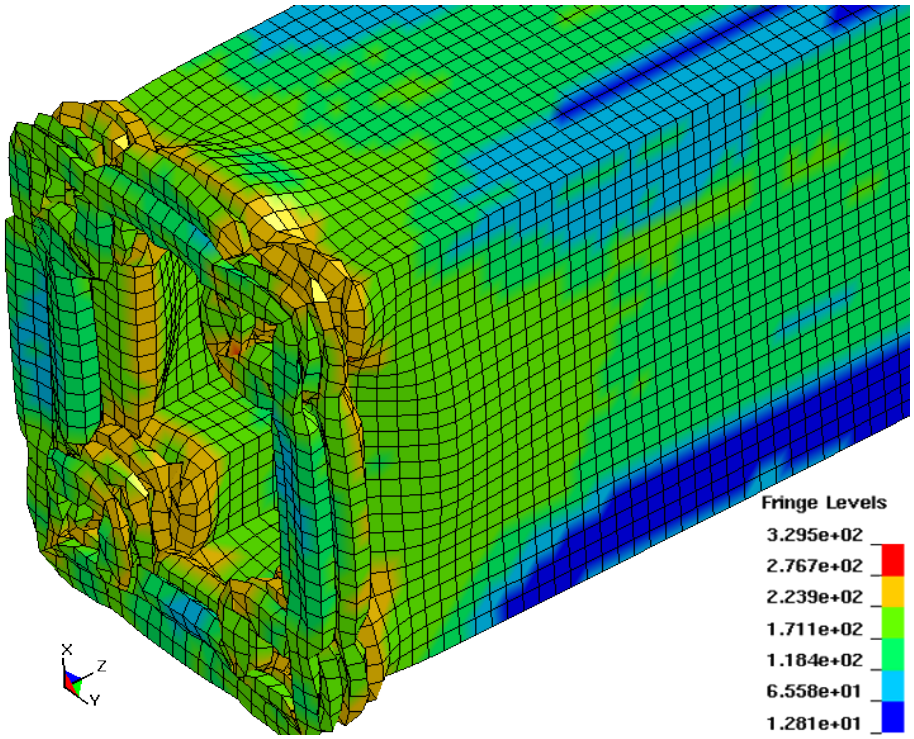


FIG. 5.43 – Best design for enhanced sizing optimisation, final deformation, von Mises stress (MPa).

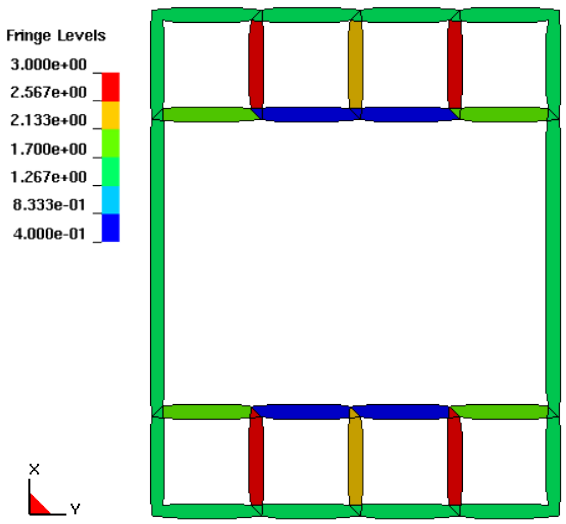


FIG. 5.44 – Best design for lower displacement constraint, thickness distribution (mm).

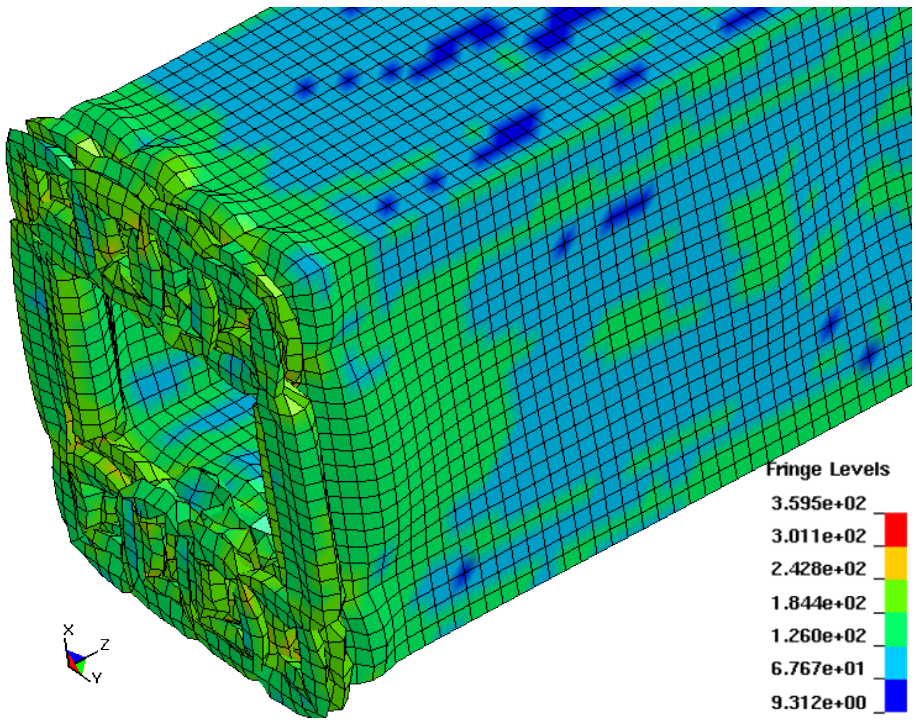


FIG. 5.45 – Best design for lower displacement constraint, final deformation, von Mises stress (MPa).

Design	Mass (g)	Displacement (mm)	SEA_{10} ($kJ.kg^{-1}$)
empty	583	111.0	2.01
cross	778	97.4	2.58
best HCATWS	878	74.7	3.77
best LS-OPT	863	74.7	3.9
best $d_{max} = 50$ mm	1125	49.9	4.08

TAB. 5.5 – Results summary for axial impact problem.

Table 5.5 summarises the results of the different designs for this problem. For easier comparison, the last column gives the Specific Energy Absorption (SEA) of each structure after 10 ms. It is defined as the energy absorbed after 10 ms divided by the mass of the structure. With this value, it is easier to see that the optimised structures are more efficient than the reference structures for energy absorption.

5.4 Third example: oblique impact

5.4.1 Problem definition

This problem is a variation on the previously defined axial impact problem. In this case, the rigid wall impacting the beam is slightly rotated around x- and y-directions (see Figure 5.46). The rotation angle in both directions is 5.7°. This angle was chosen empirically to avoid a global buckling of the empty cross-section. The mass of the rigid wall is still $M_0 = 500$ kg and the initial speed is still $v_0 = 5$ m/s, normal to the wall.

As previously, the objective is to define the reinforcement of the beam while respecting the displacement constraint d_{max} (in the impact direction). The space filling is the same as used previously, but given the load case, the symmetry condition is not applied any more. Therefore, the 31 cells are all independent now.

The same initial cross-sections are studied before running the optimisation. The empty cross-section gives a maximum displacement of 113.2 mm still with the mass of 583 g. The deformations in x- and y-directions are given in Figure 5.47 and Figure 5.48 respectively. The cross shaped reinforcement gives a maximum displacement of 101.8 mm still with the mass of 778 g. The deformations in x- and y-directions are given in Figure 5.49 and Figure 5.50 respectively. d_{max} is chosen as 80.0 mm which is 29% less than the empty cross-section and 21% less than the cross shaped reinforcement.

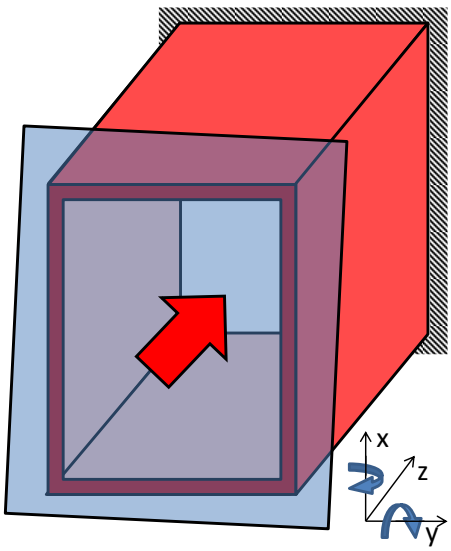


FIG. 5.46 – Oblique impact, load case.

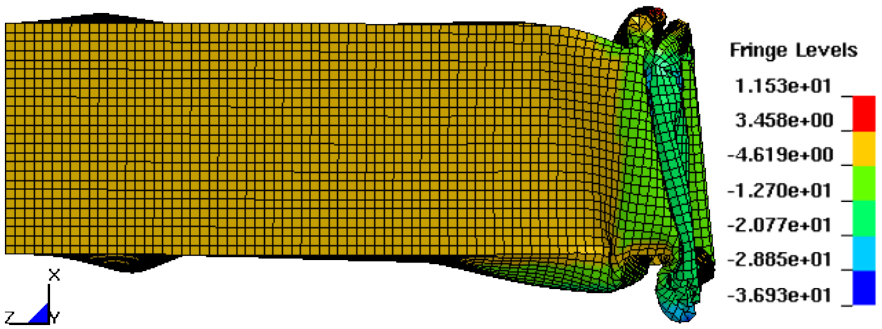


FIG. 5.47 – Oblique impact, empty cross-section, deformation in x-direction (mm).

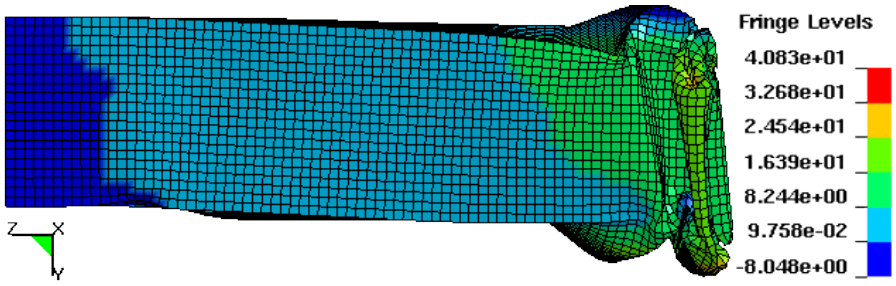


FIG. 5.48 – Oblique impact, empty cross-section, deformation in y-direction (mm).

5.4.2 Topology optimisation results

The optimisation is run with the same settings as for the axial impact. The starting thickness is set to 0.5 mm. The best design (see Figure 5.51) has a mass of 910 g (mass increase of 56.1% when compared to the empty cross-section) and a maximum displacement of 79.4 mm. The deformations in x- and y-direction (see Figure 5.52 and Figure 5.53 respectively) do not exhibit any buckling of the beam due to the oblique load. The best design is reinforced in the corner directly impacted by the wall and to a smaller extent in the two corners linked with this first corner.

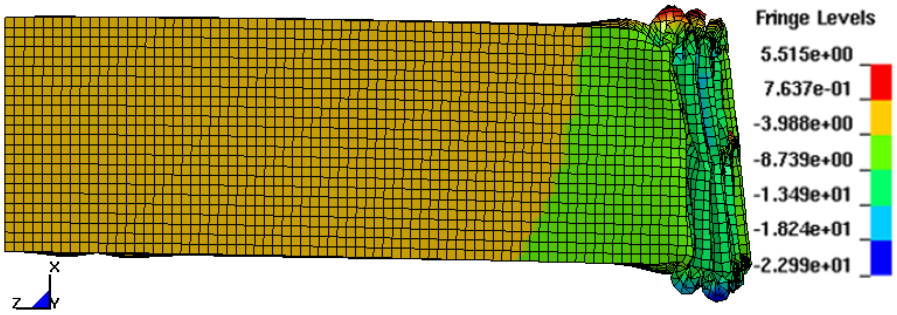


FIG. 5.49 – Oblique impact, cross reinforcement, deformation in x-direction (mm).

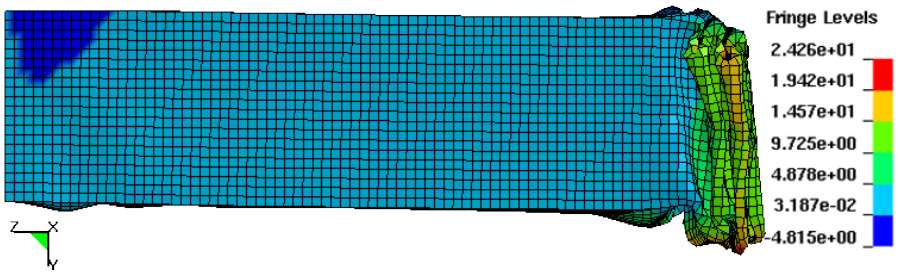


FIG. 5.50 – Oblique impact, cross reinforcement, deformation in y-direction (mm).

Only the corner impacted last by the wall is not reinforced. Notice that, on this corner a fold appears near the constrained end of the beam (see Figure 5.52 and Figure 5.53). This is a consequence of the other corners being stiffer and could lead to a global buckling. For comparison with the reference cross-sections, the force-displacement curves are given in Figure 5.54. Once again, the best design reaches higher force levels than the reference cross-sections. When compared with the axial impact (see Figure 5.39), the force levels are lower. Also, there is no clear difference between the folding initiation and propagation in the force level as it was the case for the axial impact.

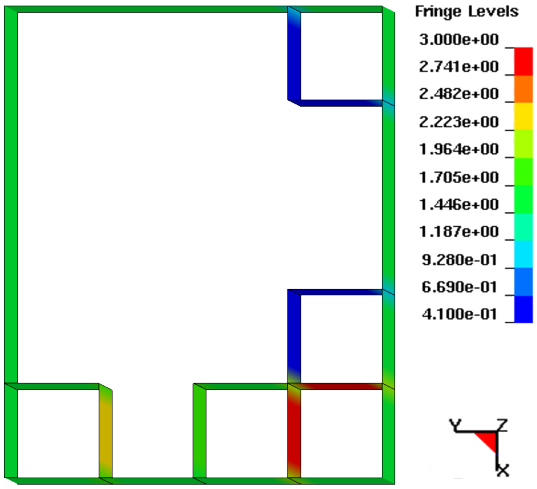


FIG. 5.51 – Oblique impact, best design, thickness distribution (mm).

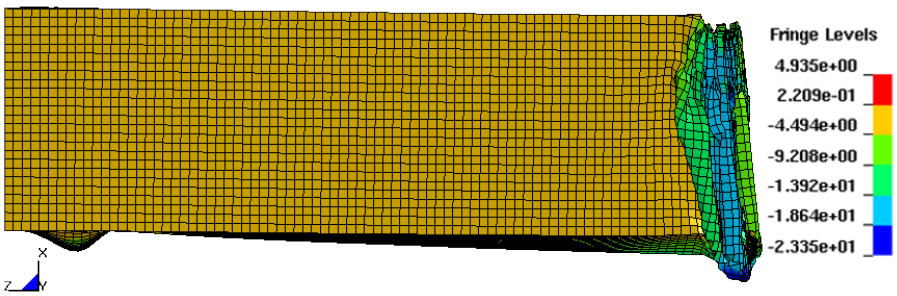


FIG. 5.52 – Oblique impact, best design, deformation in x-direction (mm).

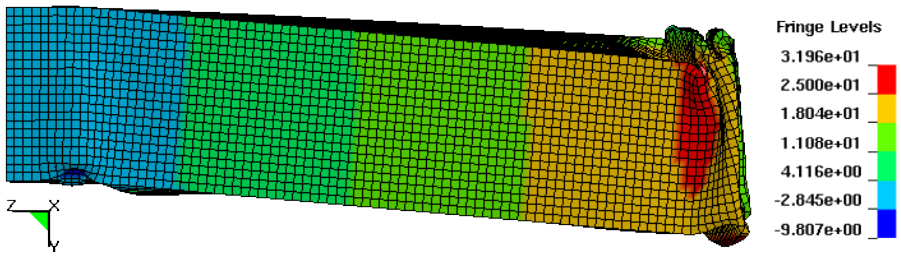


FIG. 5.53 – Oblique impact, best design, deformation in y-direction (mm).

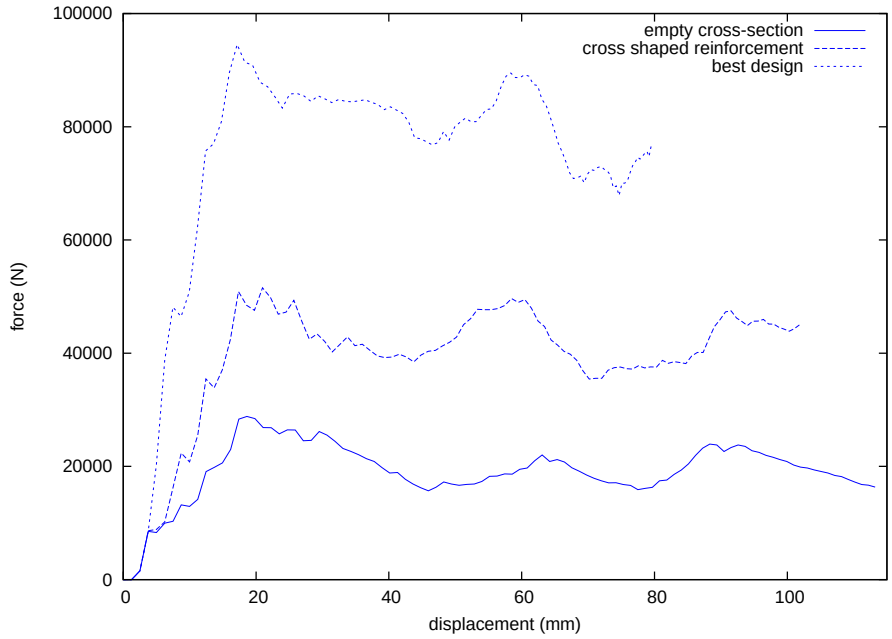
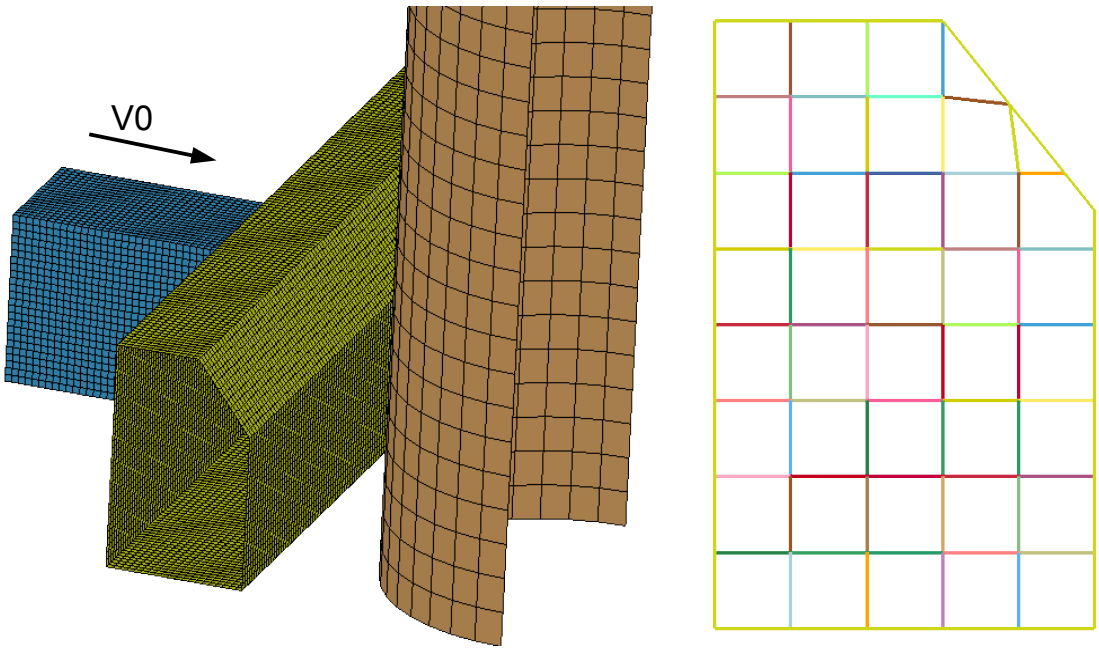


FIG. 5.54 – Oblique impact, force-displacement curves of reference cross-sections and best design (force and displacement given in impact direction).

5.5 Fourth example: simplified pole impact

5.5.1 Problem definition

This example is taken from a collaborative project on topology optimisation for crashworthiness design of extruded cross-sections [9]. The load case was defined by Ortmann et al. and is detailed in [94]. It is a simplified pole impact of a rocker and a seat cross-member (see Figure 5.55(a)). The rocker and cross-member have an initial velocity of 29 km/h and a mass of 85 kg is added at the end of the cross-member. The design task is to optimise the cross-section of the extruded rocker, while keeping the boundary fixed. The objective is to minimise the mass of the rocker while keeping the intrusion of the pole in the rocker under 75 mm and the maximum reaction force going through the cross-member under 48 kN. Notice that for the topology optimisation process, this maximum force value is only monitored and used in post-processing to dismiss the potential solutions that would violate it. The initial design is the empty cross-section with a boundary thickness of 3.5 mm. The mass is 2.801 kg, the intrusion is 69.03 mm and the maximum reaction force is 55.82 kN.



(a) Load case (b) Square space filling

FIG. 5.55 – Pole impact, problem definition.

5.5.2 Topology optimisation with square space filling

The design space is filled with a square space filling (see Figure 5.55(b)). In the diagonal area of the cross-section, the square space filling is slightly distorted to ensure good connections between the cells and the boundary. The space filling is made of 65 cells. Based on previous results from other partners of the Crash-Topo project, the thickness of the boundary is fixed to 1.75 mm. After running the optimisation, the best design found (see Figure 5.56 and Figure 5.57) has an intrusion of 74.5 mm, a mass of 1.764 kg and a maximum force of 45.2 kN.

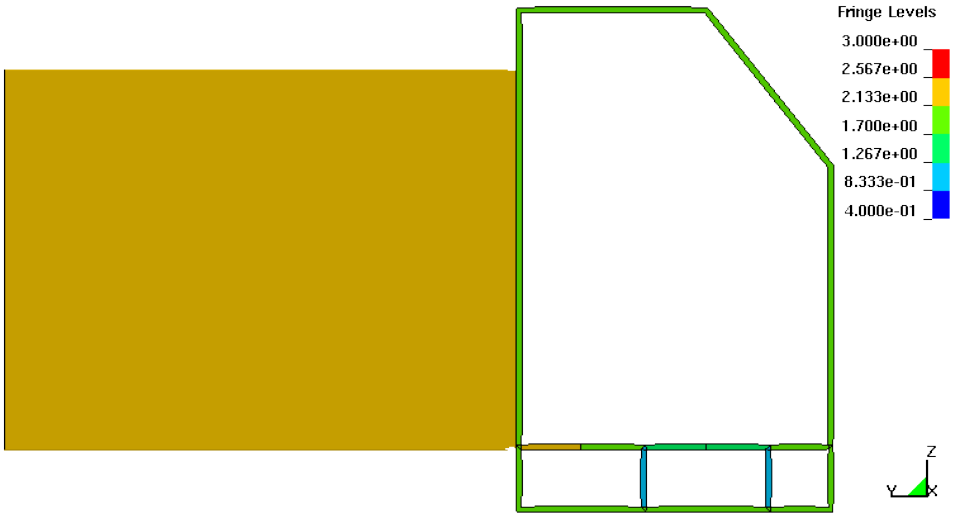


FIG. 5.56 – Pole impact, best topology, thickness distribution (mm).

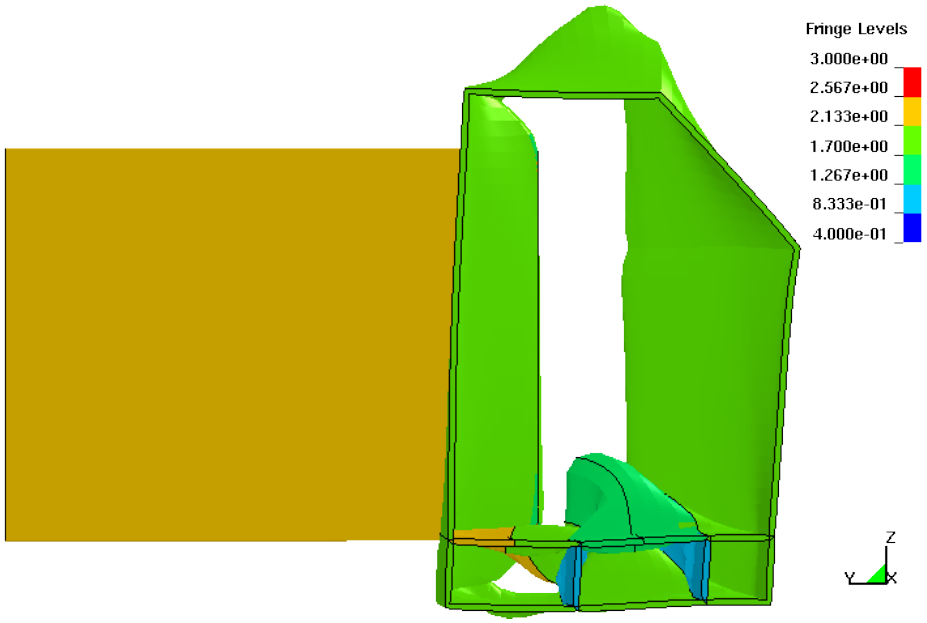


FIG. 5.57 – Pole impact, best topology, maximum deformation.

5.5.3 Additional shape optimisation

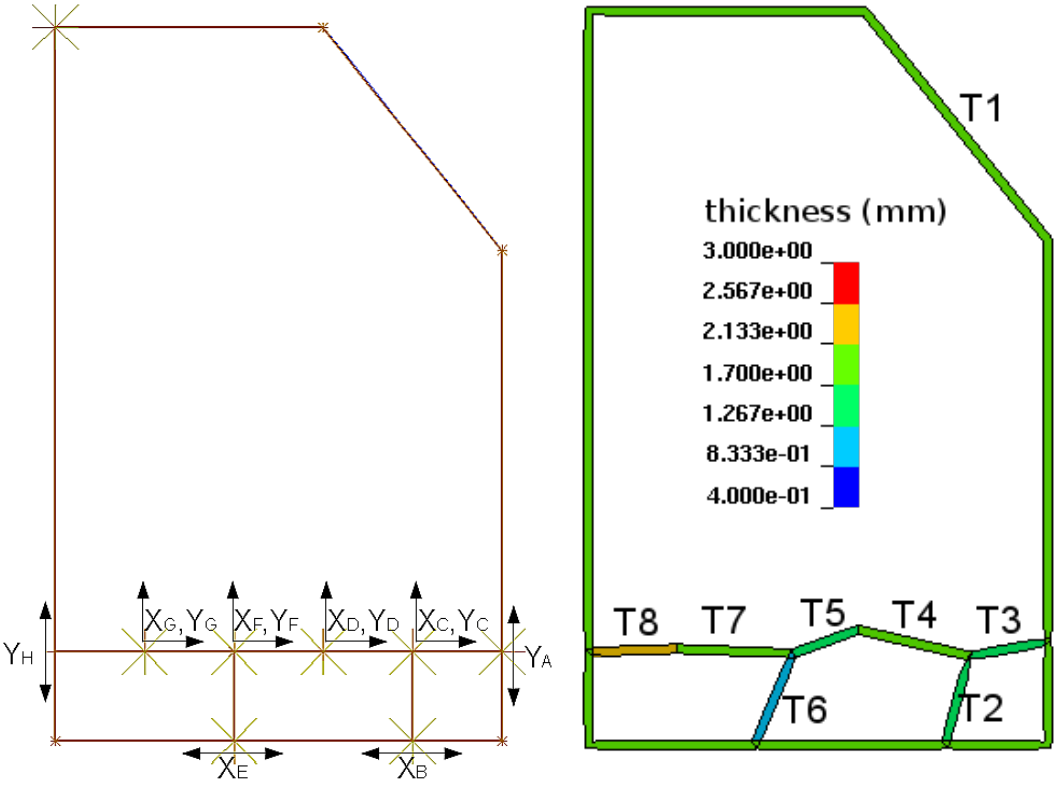
To try and improve the design further, shape optimisation is performed on the results of the topology optimisation (see Figure 5.56). This second optimisation step is performed to finely tune the results of the topology optimisation. It allows to explore designs which cannot be generated with the initial space filling of the HCATWS method.

The design parameters (see Figure 5.58(a)) are defined using the CAE software SFE CONCEPT. 20 parameters are defined with 12 translations and 8 thickness parameters. The optimisation is performed using a genetic algorithm available in LS-OPT [82]. The displacement constraint is lowered to $d_{max} = 70$ mm and the objective is now to minimise the maximum reaction force. Populations of 30 individuals are used. The best design (see Figure 5.58(b)) is found at iteration 24 with a displacement of 69.9 mm, a mass of 1.865 kg and a maximum force of 44.05 kN. Table 5.6 summarises the variations in the thickness parameters compared to the initial design. The optimised design has a more homogeneous thickness distribution with the difference between T_2 and T_3 going from 0.68 mm to 0.13 mm. Also, the difference between T_4 and T_5 goes from 0.27 mm to 0.10 mm and the difference between T_7 and T_8 goes from 0.48 mm to 0.36 mm.

The optimisation history is illustrated in Figure 5.59. After the first iteration, the reaction force increases severely in order to satisfy the modified displacement constraint. Then the force decreases with the iteration number as the algorithm finds a better geometry and thickness distribution.

Parameter	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
Initial design	1.75	1.21	1.89	1.62	1.35	1.08	1.79	2.27
Best design	1.83	1.51	1.64	1.80	1.70	1.13	1.94	2.34

TAB. 5.6 – Thickness parameters for best design and initial design (mm).



(a) Shape optimisation parameters

(b) Best design

FIG. 5.58 – Pole impact, additional shape optimisation, parameters and result.

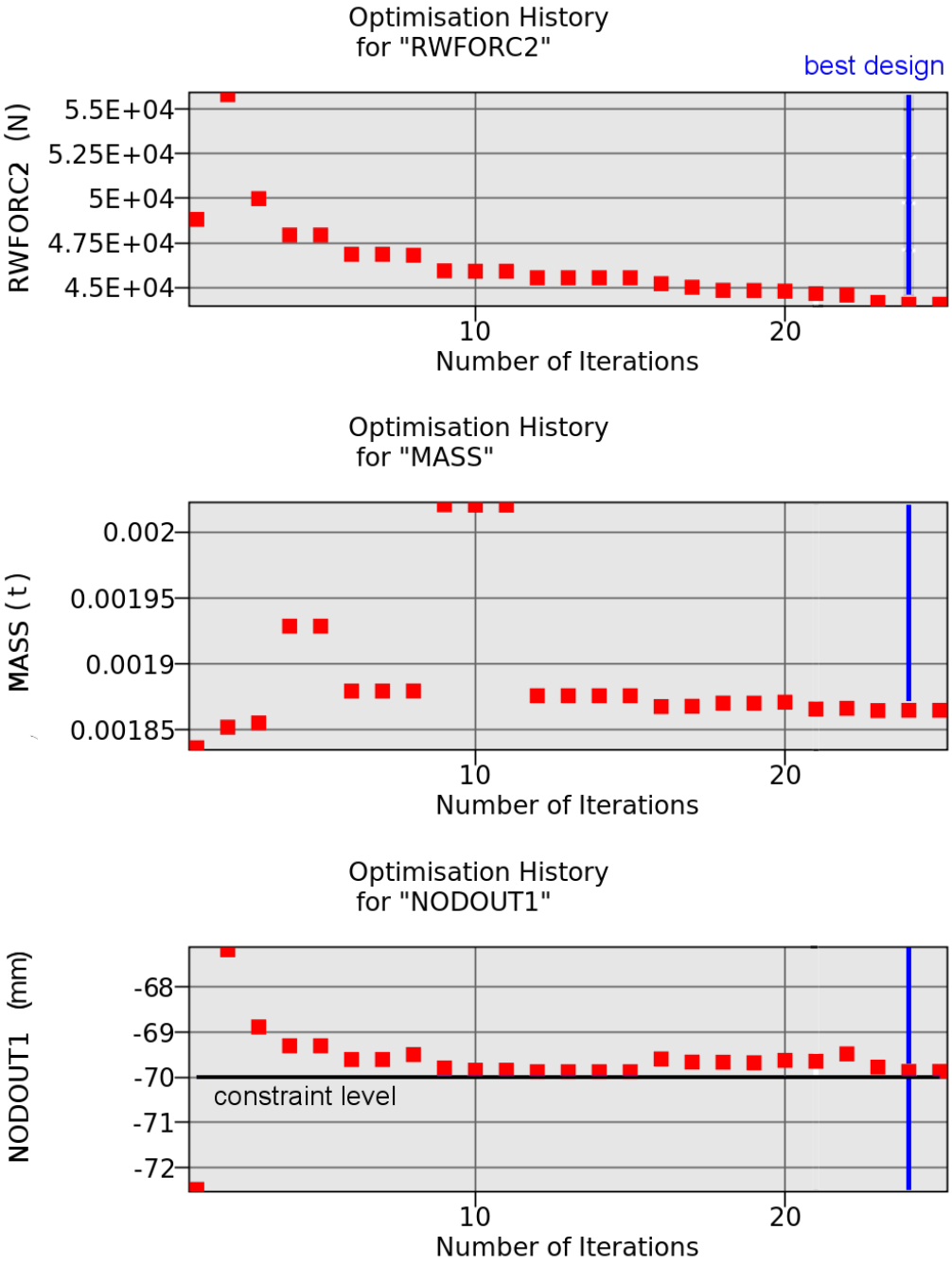


FIG. 5.59 – Pole impact, optimisation history for force objective (RWFORC2), mass (MASS) and displacement constraint (NODOUT1).

Chapter 6

Enriching the space filling with diagonal cells

Chapter 5 presented different applications of the HCATWS method for the design of extruded structures. It allowed getting a first idea of the HCATWS method capabilities in deriving credible topologies in a reasonable amount of time. It also allowed to highlight some limitations of the initial space filling. In Section 5.2, the same topology optimisation problem was solved with two different orientations of the same space filling. With the second set up, the results were significantly improved. This is a testimony that different initial space fillings will enable deriving different topologies.
In order to try and improve the HCATWS method for the design of extruded structures, a new space filling is proposed where the previously defined 2D square space filling is enriched with additional diagonal cells. In Section 6.1, the new space filling is introduced and the corresponding modifications on the algorithm are detailed. In Section 6.2, the same static shear-torsion problem as in Section 5.2 is solved with the new space filling. Similarly, Section 6.3 to Section 6.5 address the same application examples as Section 5.3 to Section 5.5. Eventually, Section 6.6 presents a transient three point bending defined to showcase the interest of coupling square and diagonal cells.

6.1 Modification of the method for diagonal enriched 2D space fillings

6.1.1 Space filling

As mentioned in introduction, the alternative space filling proposed in this chapter is based on the 2D square tiling detailed in Section 5.1.1. This space filling is enriched now with two diagonal cells in each square of the tiling (see Figure 6.1). The diagonal cells generate two additional square tilings rotated by $\pm 45°$ with re-

gards to the original tiling.
As mentioned in Section 5.1.1, the main criterion to respect for a 2D space filling in order to be used in the HCA method is to be edge-transitive (or isotoxal). The space filling proposed here is not strictly isotoxal but can still be seen as such. Indeed, although the diagonal cells are $\sqrt{2}$ times bigger than the square cells, the output value for the cells is the IED which filters the difference in dimensions. Also, each cell (from the square tiling or one of the diagonal tilings) interacts with 7 other cells at each of its extremities with the following angles: 45°, 90°, 135°, 180°, 225°, 270° and 315°. The only singularity in this pattern is the fact that within a square of the original tiling, the diagonal cells of the additional tilings cross each other. Different measures are taken to deal with this problem, as detailed in Section 6.1.2 to Section 6.1.4.

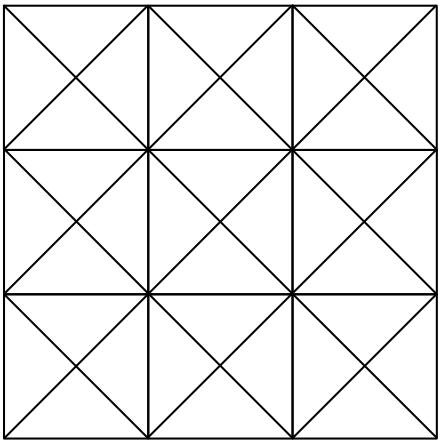


FIG. 6.1 – Diagonal enriched space filling.

6.1.2 Contact search

With the presence of intersecting cells in the initial space filling, the contact definitions become crucial. In order to regularise the space filling, the contact is defined for each diagonal cell to ignore the presence of the other diagonal cell it intersects. Note that this definition is not physical, it is artificial and allows the simulations to run with intersecting cells. As the algorithm progresses and intersecting cells are removed, the contact definition becomes physical again. Such a contact definition is not straightforward to set up and calls for specific contact search options.
A simple strategy to define contact in a model is to use a global single surface approach. In this case all the parts to be considered for contact are included in the same group and every finite element node of the group is checked for contact against every finite element. This strategy is called single surface as opposed to a surface to surface approach were a surface A is checked for contact against a surface B. While the single surface strategy is not necessarily the most efficient as it checks a lot of unnecessary pairs of nodes and finite elements, it is really easy to define and

general enough to withstand large modifications of the model. Unfortunately, the single surface approach cannot be adopted here as for each diagonal cell, a different cell needs to be excluded from the contact search. Yet, the contact definition needs to be general enough to work throughout the optimisation with the evolution of the topology.
The strategy adopted here separates the cells of the original square tiling from the diagonal cells. For the former cells, a single surface approach is adopted, including all of these non-diagonal cells. For the latter cells, a surface to surface approach is adopted with a specific definition for each of these diagonal cells. Each diagonal cell (surface A) is checked against all the other cells in the model except the diagonal cell it intersects (surface B). To be more specific, each diagonal cell is actually checked against a subset of the remaining cells defined using a triangular approach in order to limit the redundant contact searches. A simplified example of all the contact definitions used with the enriched space filling is given in Appendix B.

6.1.3 Neighbourhood

With the enriched space filling, the neighbourhood definition needs to be modified accordingly. Two approaches are investigated. In the first approach, the original square tiling and the two rotated square tilings are uncoupled while they are coupled in the second approach.

Uncoupled

In this first neighbourhood definition, the cells of the three tilings used to generate the space filling are independent. For each cell, the neighbourhood considered is the von Neumann neighbourhood for 2D thin walls defined in Section 5.1.2. It uses only cells belonging to the same initial tiling. Therefore, each non-diagonal cell possibly has two in-plane non-diagonal neighbours and possibly four out-of-plane non-diagonal neighbours. Similarly, each diagonal cell possibly has two in-plane diagonal neighbours and possibly four out-of-plane diagonal neighbours (see Figure 6.2). Non-diagonal cells do not have diagonal cells in their neighbourhood and vice-versa. This neighbourhood is uncoupled.

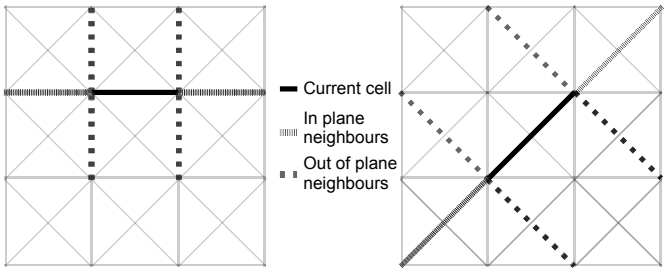


FIG. 6.2 – Uncoupled neighbourhood with diagonal cells.

Coupled

For this second neighbourhood definition, the three initial tilings are now coupled. In Section 5.1.2, the von Neumann neighbourhood contains the cells sharing an edge (or a node in the 2D representation). Using this definition, the coupled neighbourhood of a non-diagonal cell contains not only two in-plane and four out-of-plane neighbours, but also eight diagonal neighbours (see Figure 6.3). Four of these diagonal neighbours generate a $\pm 45°$ angle with the non-diagonal cell and are therefore referred to as near diagonal neighbours. The remaining four neighbours generate a $\pm 135°$ angle with the non-diagonal cell and are therefore referred to as far diagonal neighbours. Similarly, each diagonal cell has not only two in-plane and four out-of-plane neighbours, but also eight non-diagonal neighbours which in the diagonal cell's respect are near and far diagonal neighbours.

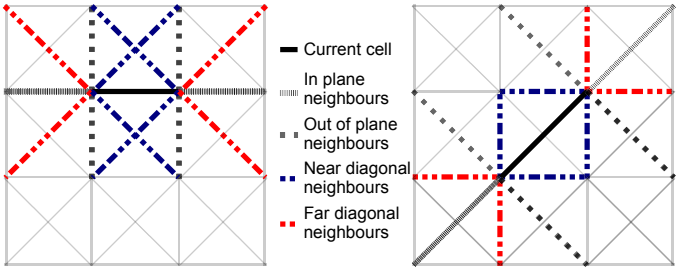


FIG. 6.3 – Coupled neighbourhood with diagonal cells.

6.1.4 Algorithm

With the modifications in the space filling and neighbourhood definition, the algorithm, and especially the update rule needs to be adapted. For the uncoupled neighbourhood, the Hybrid update rule described in Section 4.3.4 is applied to both the non-diagonal and the diagonal cells.
In order to generate topologies free of intersecting diagonal cells, an additional rule is introduced and applied after the Hybrid update rule. If n and m are active intersecting diagonal cells, the additional rule increases the difference in their thickness in order to favour the deletion of at least one of these two cells. The new thickness values are defined as:

$$\begin{aligned} & t'_n = max\left(t_{min}, min\left(t_{max}, t_n + \left(t_n - t_m\right)/2\right)\right). \\ \text{and} \quad & t'_m = max\left(t_{min}, min\left(t_{max}, t_m + \left(t_m - t_n\right)/2\right)\right). \end{aligned} \tag{6.1}$$

For coupled neighbourhood, the Hybrid update rule is extended to account for the diagonal neighbours. For each diagonal neighbour, an extra α coefficient is defined and added to the thickness variation $\delta t_i^{(j,k)}$ (see Equation (4.8)). For near diagonal

neighbours, the coefficient $\alpha_{nd}^{(j,k)}$ is defined as:

$$\begin{aligned}
\alpha_{nd}^{(j,k)} = & \sum_{g=1}^{l_{nd}} \zeta \times \mathbf{1}_{P_g}\left(\tilde{U}_{nd}^{(k)}\right) - \sum_{g=1}^{l_{nd}} \zeta \times \mathbf{1}_{M_g}\left(\tilde{U}_{nd}^{(k)}\right). \\
\text{where} \quad P_g = & \left[S^{*(j,k)} + b_{nd,g} \times \text{disc}_{\text{up}}^{(j,k)};\ S^{*(j,k)} + \text{disc}_{\text{up}}^{(j,k)}\right]. \\
M_g = & \left[S^{*(j,k)} - \text{disc}_{\text{low}}^{(j,k)};\ S^{*(j,k)} - b_{nd,g} \times \text{disc}_{\text{low}}^{(j,k)}\right]. \\
\text{and} \quad & \forall 1 \leq g < l_{nd}, \quad 0 < b_{nd,g} < b_{nd,g+1} < 1.
\end{aligned} \tag{6.2}$$

For far diagonal neighbours, the coefficient $\alpha_{fd}^{(j,k)}$ is defined as:

$$\begin{aligned}
\alpha_{fd}^{(j,k)} = & \sum_{h=1}^{l_{fd}} \zeta \times \mathbf{1}_{P_h}\left(\tilde{U}_{fd}^{(k)}\right) - \sum_{h=1}^{l_{fd}} \zeta \times \mathbf{1}_{M_h}\left(\tilde{U}_{fd}^{(k)}\right). \\
\text{where} \quad P_h = & \left[S^{*(j,k)} + b_{fd,h} \times \text{disc}_{\text{up}}^{(j,k)};\ S^{*(j,k)} + \text{disc}_{\text{up}}^{(j,k)}\right]. \\
M_h = & \left[S^{*(j,k)} - \text{disc}_{\text{low}}^{(j,k)};\ S^{*(j,k)} - b_{fd,h} \times \text{disc}_{\text{low}}^{(j,k)}\right]. \\
\text{and} \quad & \forall 1 \leq h < l_{fd}, \quad 0 < b_{fd,h} < b_{fd,h+1} < 1.
\end{aligned} \tag{6.3}$$

In both cases, notations are the same as in Section 4.3.4. As for the uncoupled neighbourhood, an additional update is applied in order to get rid of the intersecting cells. It uses the same definition (see Equation (6.1)).

6.2 Shear-torsion case

6.2.1 Problem definition

This shear-torsion static problem is already defined in Section 5.2. Again, the design space is filled with cells, but this time using the diagonal enriched space filling defined above (see Figure 6.4). With a base dimension of 30×90 mm for the non-diagonal cells, the space filling is made of 118 cells (compared to 52 with the initial square space filling of same dimensions and 56 with the rotated square space filling).

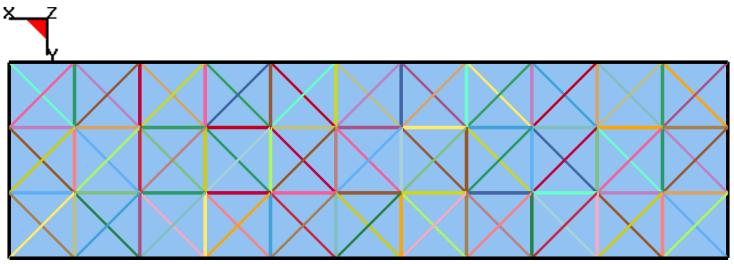


FIG. 6.4 – Diagonal enriched space filling for shear-torsion case.

6.2.2 Optimisation results

For this problem, a coupled neighbourhood is used. The uncoupled neighbourhood is investigated in Section 6.3. The optimisation is run with three different initial mass fractions: 0.5, 0.6 and 0.8. This value was kept voluntarily low as the enriched space filling generates a heavy reference structure. The other parameters are kept constant for all runs and are given in Table 6.1. The only new parameters here are the update rule parameters associated with the diagonal neighbours, l_{nd}, $b_{nd,1}$, l_{fd} and $b_{fd,1}$. $b_{nd,1}$ is chosen bigger than $b_{fd,1}$ in order to limit the influence of the near diagonal neighbours in the update rule. This is meant to limit the smaller angles ($\pm 45°$) in the final topology. Also, reintroduction of cells is not enforced here, but to favour spontaneous reintroduction of cells, deleted cells do not participate in their own thickness update, only their active neighbours do.
The best design was found for $M_{f,0} = 0.6$ with a mass of 769 g (or a mass increase of 3.5% compared to the empty beam). 9 cells remain in the space filling (see Figure 6.5). They form a similar topology to the one derived with the rotated square space filling (see Section 5.2.3). Evolution of the structure mass and of the number of active cells in the model are given in Figure 6.6 and Figure 6.7. Figure 6.6 shows that the best design is under 800 g after only 10 iterations. Figure 6.7 shows the deletion and reintroduction of cells during the optimisation. Because the reintroduction of cells is not enforced, only a limited number of cells get reintroduced. This favours the convergence of the algorithm.

$t_{\min}$	$t_{\max}$	$k_{\max}$	ε_1	ε_2	$\delta t_{\max}$
0.4 (mm)	3.0 (mm)	200	1e-4	0.1 (mm)	0.5 (mm)
η	τ	θ	l_q	$b_{q,1}$	$b_{q,2}$
0.3	50	3	2	0.1	0.5
l_0	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$	$b_{0,5}$
5	0.05	0.1	0.2	0.4	0.7
l_r	$b_{r,1}$	l_{nd}	$b_{nd,1}$	l_{fd}	$b_{fd,1}$
1	0.1	1	0.6	1	0.3

TAB. 6.1 – Common parameters for three optimisation runs.

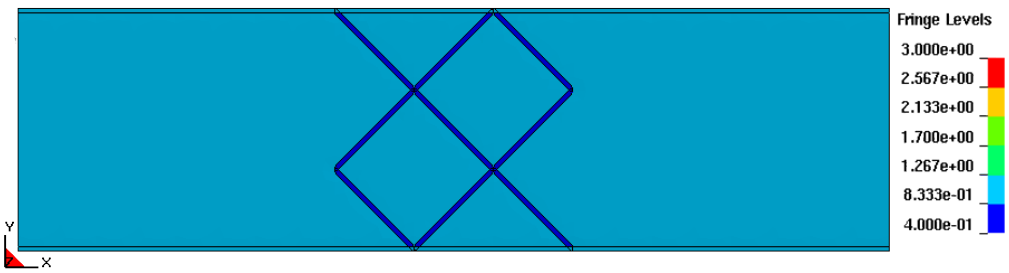


FIG. 6.5 – Shear-torsion case, best design with diagonal enriched space filling, thickness distribution (mm).

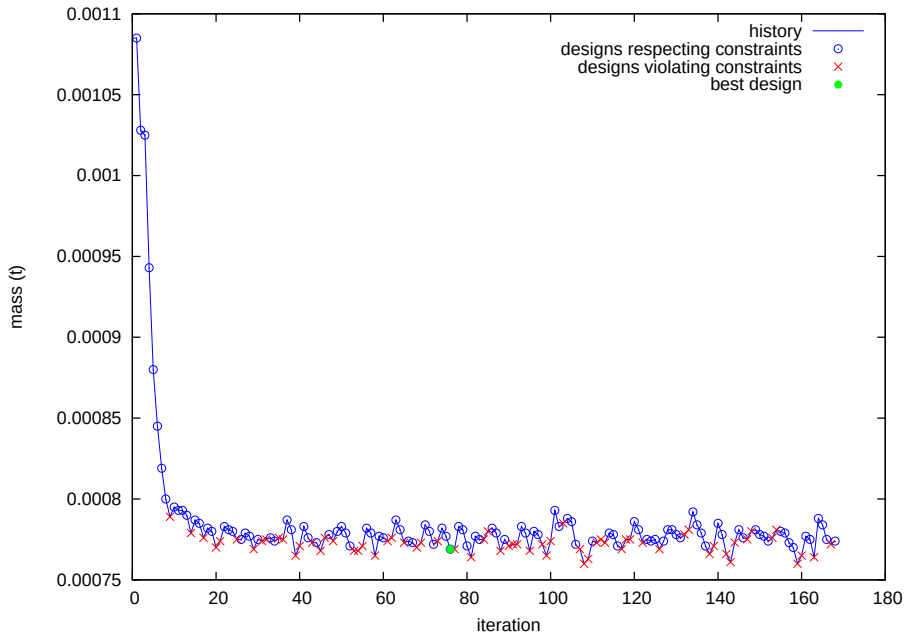


FIG. 6.6 – Shear-torsion case, optimisation history of diagonal enriched space filling.

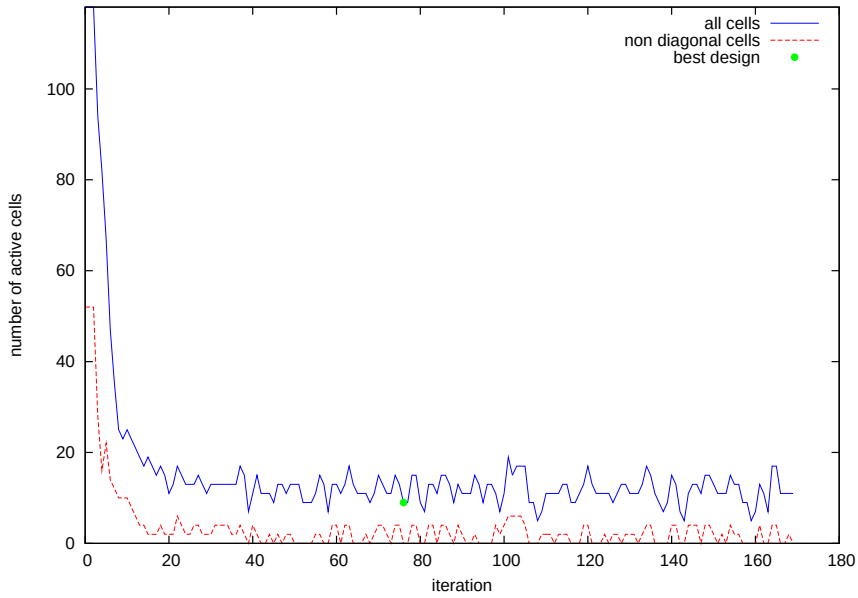


FIG. 6.7 – Shear-torsion case, evolution of active cells for diagonal enriched space filling.

6.3 Axial impact

6.3.1 Problem definition

The problem is the same as defined in Section 5.3. The space filling used previously is enriched with diagonal cells (see Figure 6.8). The number of cells is 71, as compared to 31 with the square space filling. The problem is solved both with uncoupled (see Section 6.3.2) and coupled (see Section 6.3.3) neighbourhoods

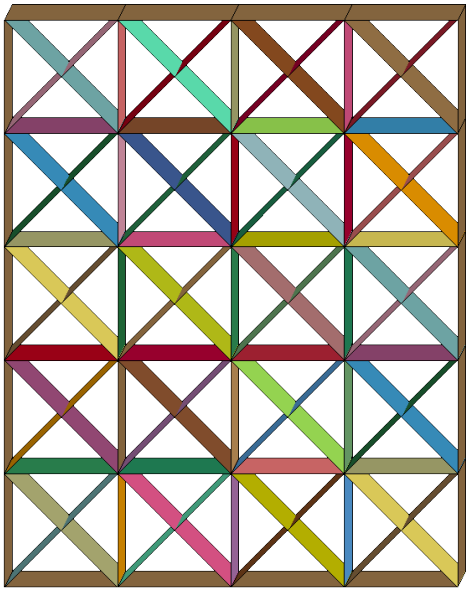


FIG. 6.8 – Diagonal enriched space filling of the cross-section for axial impact.

6.3.2 Results with uncoupled neighbourhood

The problem is first solved with the uncoupled neighbourhood described in Section 6.1.3. The parameters are similar to the ones used for the previous problem (see Table 6.1). To reduce optimisation time, the maximum number of iterations $k_{\max}$ is set to 100 and the maximum mass fraction change allowed is set to fade quickly with $\eta = 0.6$ and $\tau = 10$. Initial mass fraction $M_{f,0}$ is set to 0.45. With the uncoupled neighbourhood, the diagonal neighbours update rule parameters defined in Table 6.1 do not apply. Also, in this case, reintroduction of cells is enforced as described in Section 4.3.4.

The best design is found at iteration 31 with a mass of 1067 g and a displacement of 74.6 mm (see Figure 6.9). With 47 out of 71 cells remaining, the topology is not clearly defined and the mass value is much higher than the 878 g derived for the same problem using the square space filling (see Section 5.3.2). Also, four crossing diagonal cells remain in the model which is not desirable.

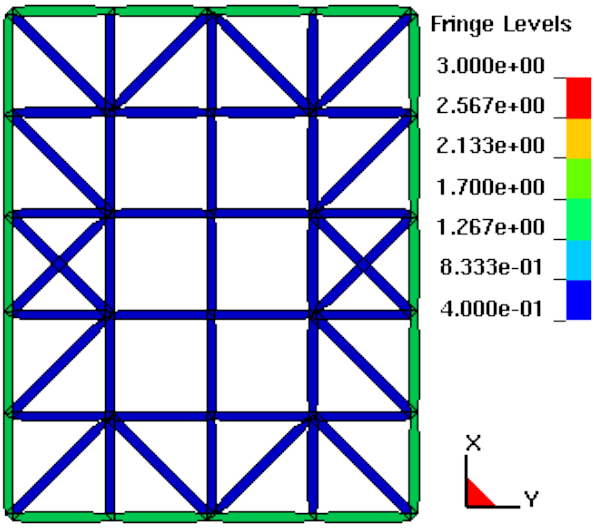


FIG. 6.9 – Axial impact, best design for uncoupled neighbourhood (iteration 31), thickness distribution (mm).

6.3.3 Results with coupled neighbourhood

The problem is now solved with the coupled neighbourhood described in Section 6.1.3. For comparison, the parameters are the same as for the uncoupled neighbourhood. The additional diagonal neighbours update rule parameters are defined as in Table 6.1.

The best design is found at iteration 81 with a mass of 930 g and a displacement of 72.6 mm. This is much better than the result derived with the uncoupled neighbourhood and closer to the results derived with the square space filling. Also, with only 18 cells remaining, the topology is more clearly defined (see Figure 6.10) and really similar to the topology derived with the square space filling (see Section 5.3). This suggests that the coupled neighbourhood is more adapted to this diagonal enriched space filling than the uncoupled neighbourhood.

In a second optimisation run, the initial mass fraction is slightly modified ($M_{f,0} = 0.6$) and the maximum number of iterations is increased to $k_{\max} = 200$. All other parameters remain the same. This time, the best design is found at iteration 188 with a mass of 883 g and a displacement of 72.8 mm. The topology is made of 16 cells, with no diagonal cells remaining (see Figure 6.11). As for the square space filling (see Figure 5.35), the rectangular beam is reinforced in its corners.

Figure 6.12 to Figure 6.14 illustrate the evolution of active cells for, respectively, the best optimisation run with square space filling (see Section 5.3.2), the first and second runs with diagonal enriched space filling and coupled neighbourhood. Similarly, Figure 6.15 to Figure 6.17 illustrate the mass history for the same optimisation runs.

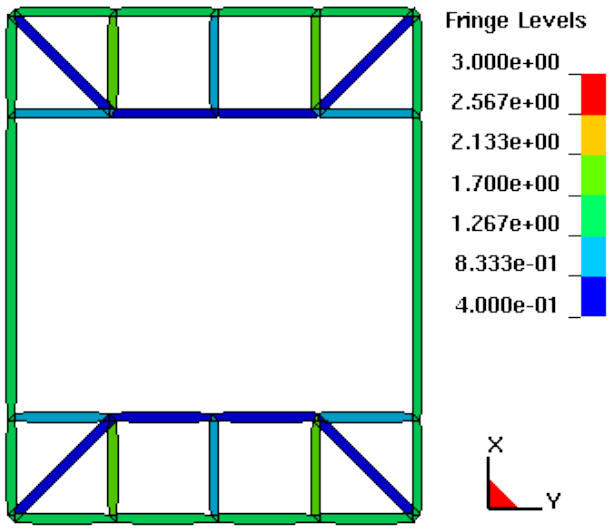


FIG. 6.10 – Axial impact, best design for coupled neighbourhood (iteration 81), thickness distribution (mm).

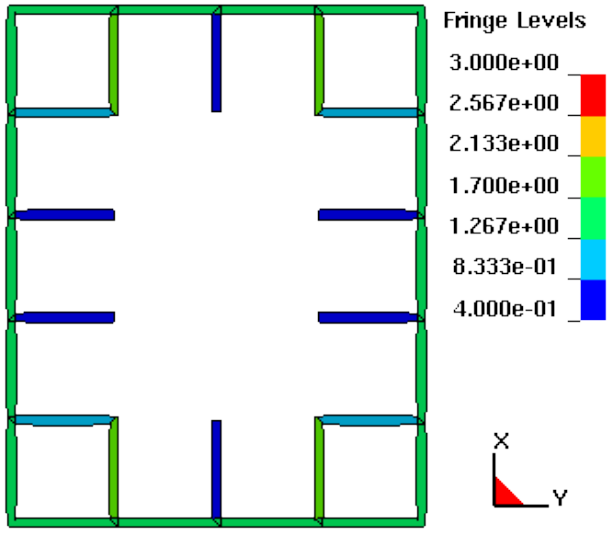


FIG. 6.11 – Optimum design topology third run, walls thickness distribution (mm).

Whether it is the number of active cells or the mass of the model, large oscillations can be observed for the two optimisation runs with diagonal enriched space fillings. This is a consequence of the enforced reintroduction used here. Indeed, the algorithm tends to reintroduce most of the deleted cells at once when the reintroduction happens. This seems to have a negative effect on the convergence of the algorithm and would suggest that the reintroduction setting used to solve the shear-torsion problem (see Section 6.2.2, no reintroduction enforced and no contribution of the deleted cells to their own update) is more efficient with higher number of cells in the model.

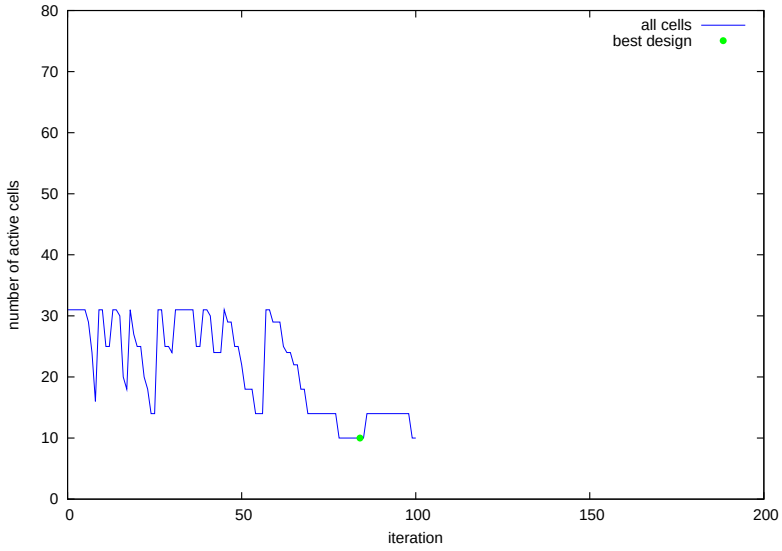


FIG. 6.12 – Axial impact, active cells for square space filling 2nd run.

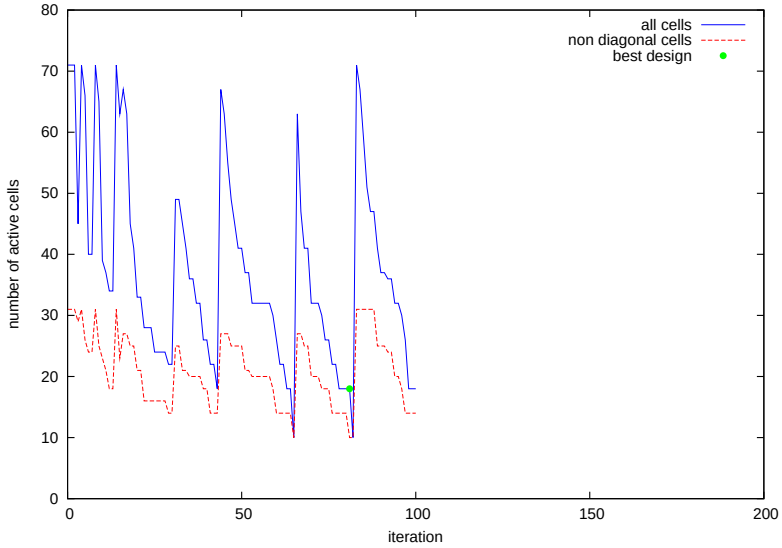


FIG. 6.13 – Axial impact, active cells for coupled diagonal space filling, 1st run.

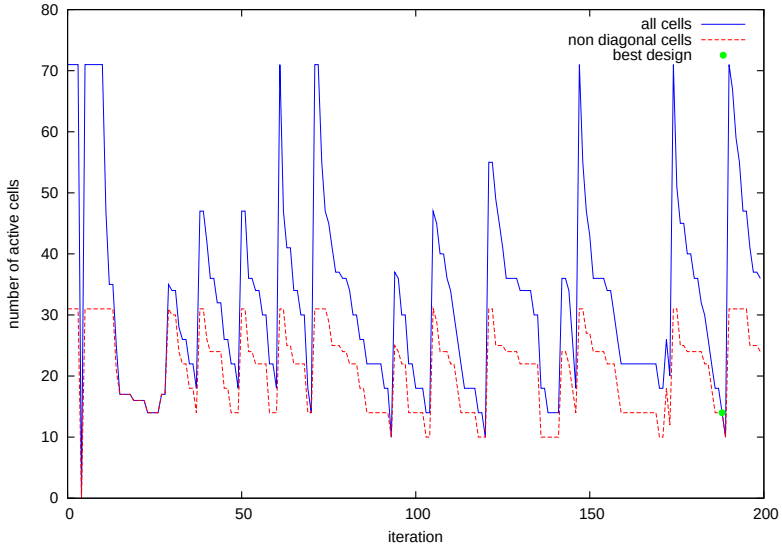


FIG. 6.14 – Axial impact, active cells for coupled diagonal space filling, 2nd run.

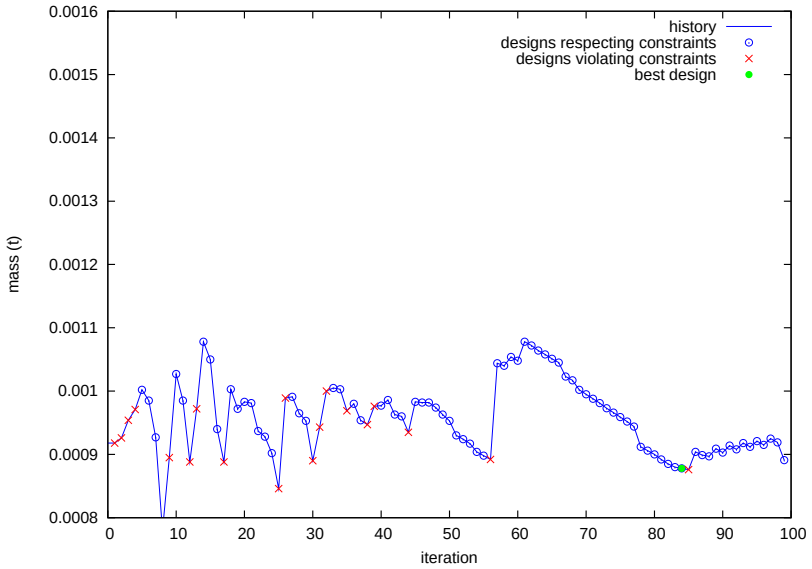


FIG. 6.15 – Axial impact, mass history for square space filling 2nd run.

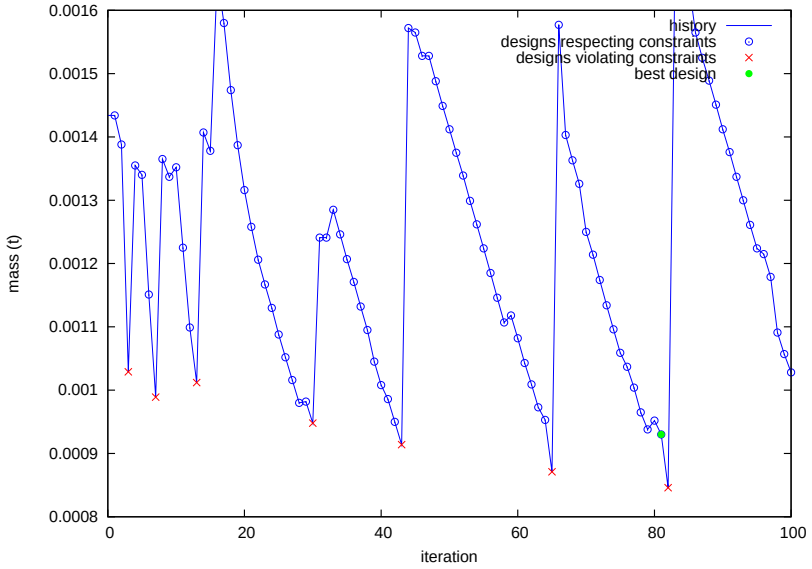


FIG. 6.16 – Axial impact, mass history for coupled diagonal space filling, 1st run.

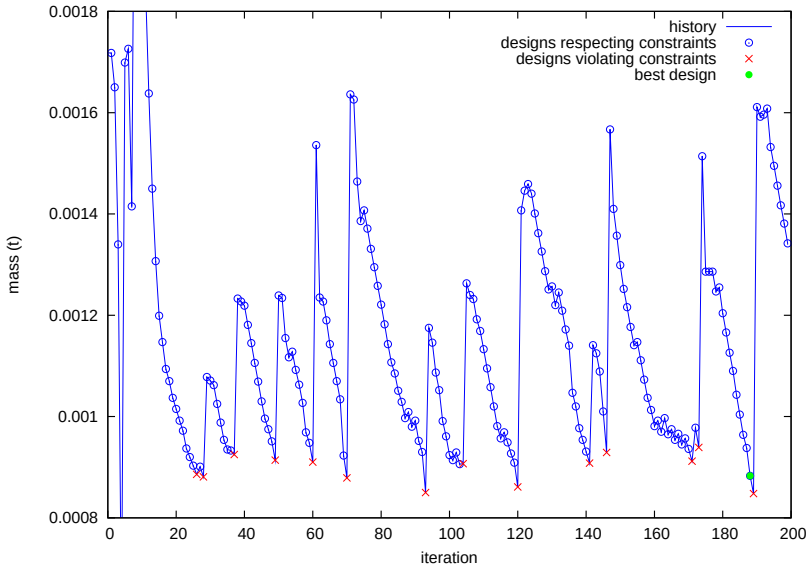


FIG. 6.17 – Axial impact, mass history for coupled diagonal space filling, 2nd run.

6.4 Oblique impact

6.4.1 Problem definition

The problem studied here is the same as the one defined in Section 5.4 (see Figure 5.46). The design space being the same as for the axial impact problem, the enriched space filling is the same as in Figure 6.8. Again, it is made of 71 cells instead of 31 for the square space filling. Given the results on the axial impact problem, only the coupled neighbourhood is used here.

6.4.2 Optimisation results

The problem is solved with different settings. In the first two runs, the same parameters as the ones used for the axial impact and coupled neighbourhood are used (see Section 6.3.3). In both cases, $k_{\max} = 200$ and the initial mass fraction is set to 0.5 in the first run and 0.7 in the second run. In a third run, the reintroduction of cells is not enforced any more and the initial mass fraction is set to 0.9.
The best design is found in the third run at iteration 146 with a mass of 910 g and a displacement of 79.9 mm. Although it is the same mass value as the best design derived from the square space filling, the topology is different (see Figure 6.18). Figure 6.19 illustrates the optimisation history of the third run. With no reintroduction enforced, no big oscillations as observed in the axial impact results (see Figure 6.16 and Figure 6.17) appear. Figure 6.20 and Figure 6.21 illustrate the deformation of the best design. With cross-section reinforced in only one corner, the beam buckles under the oblique load. This is not desirable in general, but within the definition of the optimisation, such a phenomenon was not avoidable. To avoid it, a more advanced constraint should be defined.

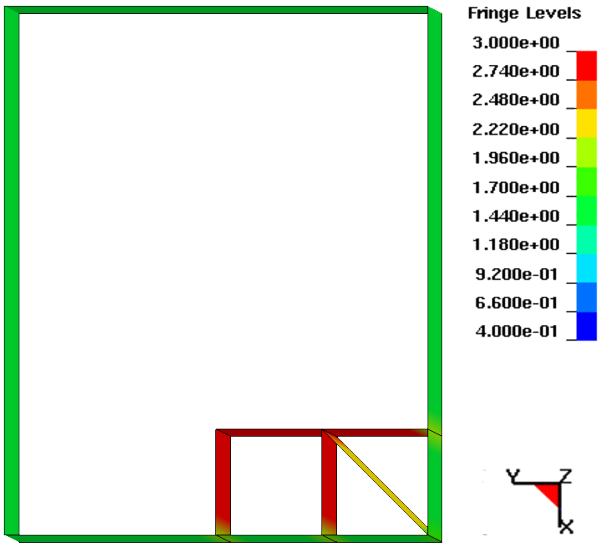


FIG. 6.18 – Optimum design topology for oblique impact, thickness distribution (mm).

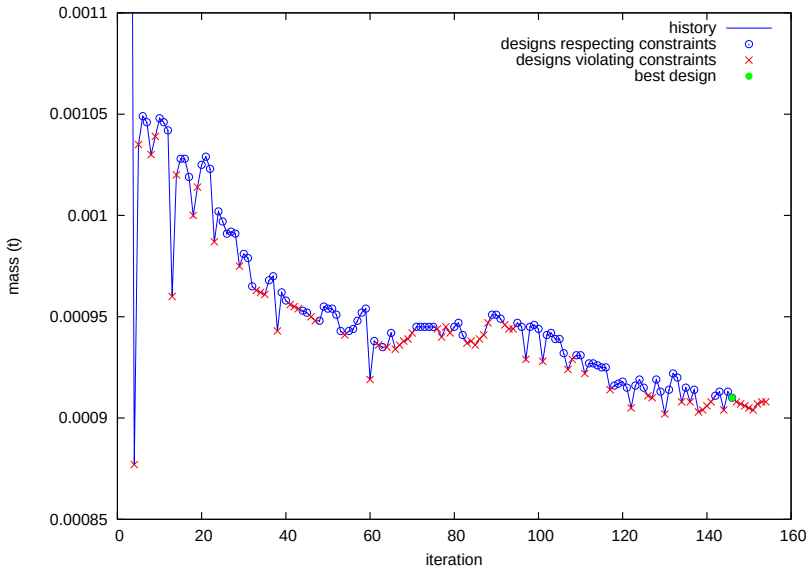


FIG. 6.19 – Oblique impact, mass history for coupled diagonal space filling, 3$^{\text{rd}}$ run.

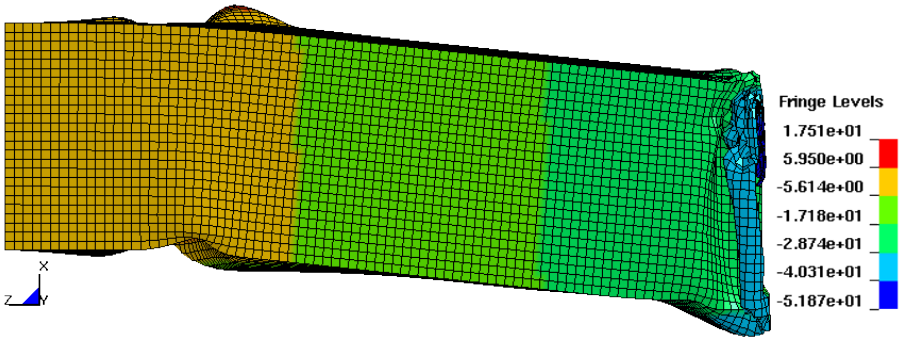


FIG. 6.20 – Oblique impact, best design, deformation in x-direction (mm).

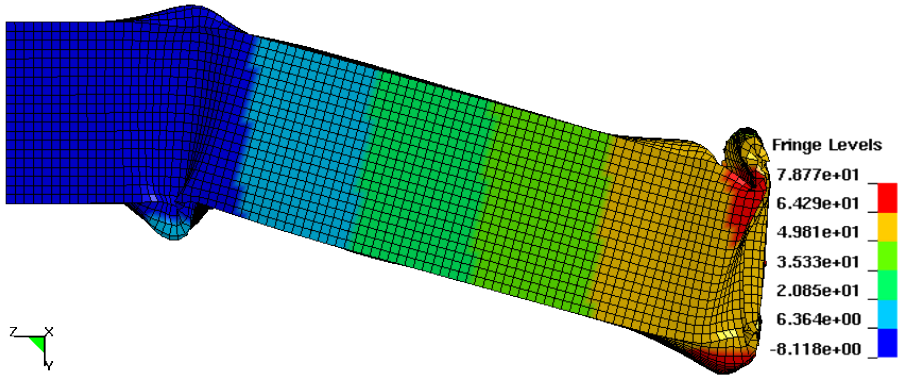


FIG. 6.21 – Oblique impact, best design, deformation in y-direction (mm).

6.5 Pole impact

6.5.1 Problem definition

This problem is already defined in Section 5.5.1. The enriched space filling is made of 139 cells where the square space filling was made of only 65 cells. Again, with the more complex design space, the cells are distorted in the top right corner (see Figure 6.22).

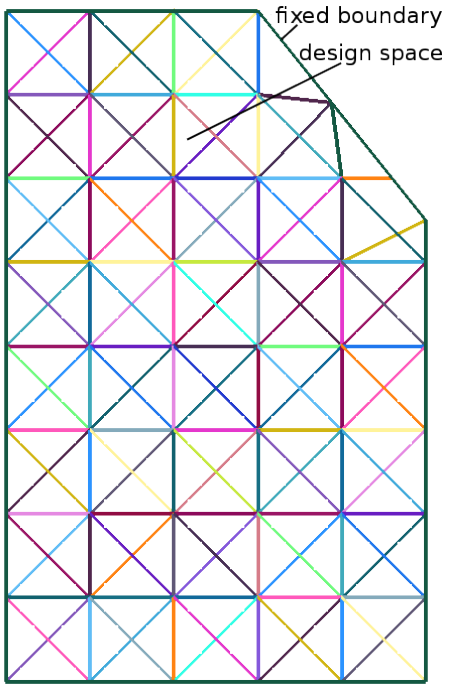


FIG. 6.22 – Pole impact, diagonal enriched space filling.

6.5.2 Topology optimisation result

The optimisation problem is solved with different values for the outer boundary wall thickness. The values studied are 1.5, 1.7, 1.75, 1.8, 1.85 and 2.0 mm. In all cases, the initial mass fraction for the reinforcements is set to 0.5, the neighbourhood is coupled and the reintroduction of cells is not enforced. The other optimisation parameters are defined in Table 6.2. The best result is found for the 1.75 mm thick boundary wall (see Figure 6.23) with an intrusion of 71.4 mm, a mass of 1.951 kg and a maximum force of 46.3 kN. The topology is more complex than the one generated with the square space filling (see Figure 5.56) and the mass is higher. In this case, the square space filling gives a better result.

$t_{\min}$	$t_{\max}$	$k_{\max}$	ε_1	ε_2	$\delta t_{\max}$
0.4 (mm)	3.0 (mm)	200	1e-4	0.1 (mm)	0.5 (mm)
η	τ	θ	l_q	$b_{q,1}$	$b_{q,2}$
0.05	50	3	2	0.1	0.5
l_0	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$	$b_{0,5}$
5	0.05	0.1	0.2	0.4	0.7
l_r	$b_{r,1}$	l_{nd}	$b_{nd,1}$	l_{fd}	$b_{fd,1}$
1	0.1	1	0.6	1	0.3

TAB. 6.2 – Common parameters for all optimisation runs.

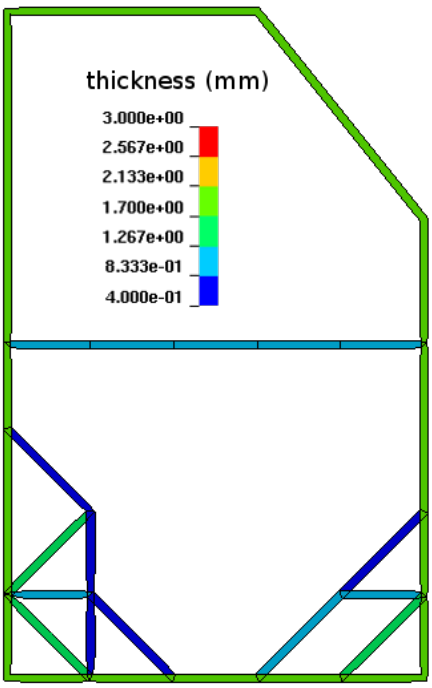
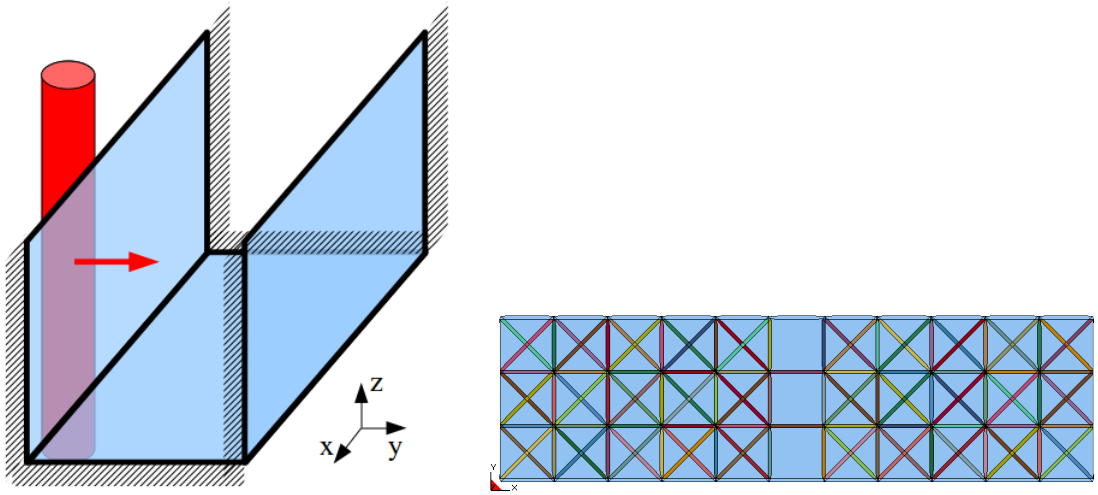


FIG. 6.23 – Pole impact, best topology for diagonal enriched space filling, thickness distribution (mm).

6.6 Three point bending

6.6.1 Problem definition

Another application example is defined here. It is inspired by a design case presented in [58] (see Figure 3.4), where an impactor hits a rectangular beam constrained at both ends in its centre. A similar problem is also studied in [10]. In the problem studied here, the beam has the same dimensions and characteristics as for the shear-torsion case defined in Section 5.2. It is impacted by a pole with an initial velocity in y-direction (see Figure 6.24(a)).
As for the shear-torsion case, the objective of the problem is to derive an adequate topology to reinforce the beam. This time though, the reinforcement is made of the same material as the beam. The material parameters are the same as for the axial impact problem (see Section 5.3). The enriched space filling excludes central diagonal cells for a total of 112 cells (see Figure 6.24(b)). Given the symmetry of the problem, keeping these central diagonal cells could generate crossing cells in the final topology. The design constraint is defined on the maximum impactor displacement in y-direction, $d_{max} = 45$ mm. In the next section, the influence of the impact velocity on the derived topology is investigated.



(a) Load case (b) Enriched space filling

FIG. 6.24 – Three point bending, problem definition.

6.6.2 Topology optimisation with different impact velocities

In order to assess the influence of the crash non-linearities on the derived topology, different impactor velocities are tested. The low speed impact is set to 1 m/s, the intermediate speed is set to 10 m/s and the high speed is set to 50 m/s. The mass of the impactor is adjusted in all cases in order to have the same initial kinetic energy in all cases.

The optimisation is run with the same parameters as for the shear-torsion case (see Table 6.1) and an initial mass fraction of 0.5 for all three cases. Again, the reintroduction of cells is not enforced and the deleted cells do not contribute to their own thickness update. Table 6.3 summarises the results for all three cases.

Velocity (m/s)	Mass (g)	Displacement (mm)	remaining cells
1	473	44.7	24
10	440	44.9	19
50	378	44.5	20

TAB. 6.3 – Results summary three point bending problem.

The topology, the deformed state and the optimisation history are given in Figures 6.25 to 6.27 for the low velocity impact, Figures 6.28 to 6.30 for the intermediate velocity impact and Figures 6.31 to 6.33 for the high velocity impact. Although the energy to absorb is the same in all three cases, the mass needed to absorb it is higher for the low velocity impact and lower for the high velocity impact. This clearly indicates the influence of the dynamic effects. Also, the fact that the topology derived for the low velocity impact differs from the intermediate and high velocity impacts is really interesting as it questions the use of Equivalent Static Loads to perform topology optimisation of high deformation load cases.
In terms of algorithm, in all cases, a steady improvement of the best design can be observed with no big oscillations as in other previous cases. This is a good indication of the quality of the optimisation parameters chosen here.

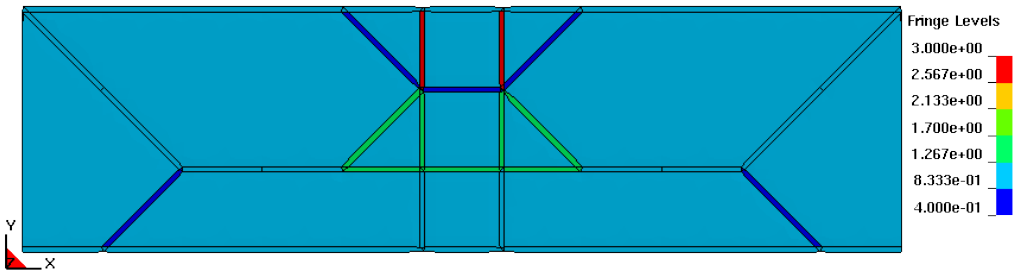


FIG. 6.25 – Three point bending, best design for low velocity, thickness distribution (mm).

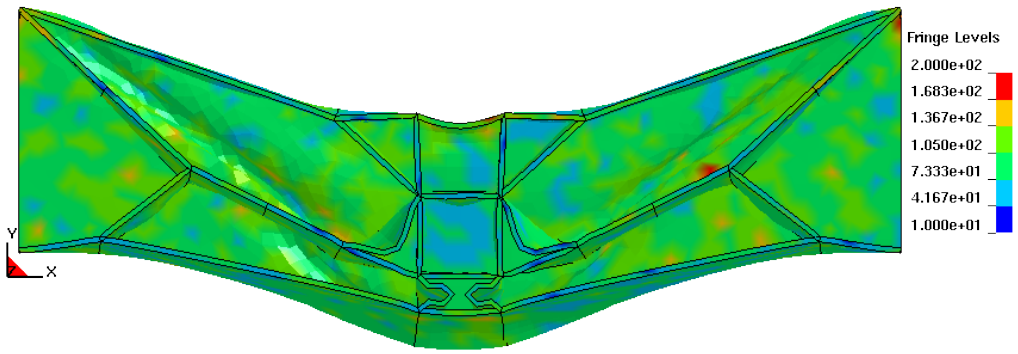


FIG. 6.26 – Three point bending, best design for low velocity, von Mises stress under deformation (MPa).

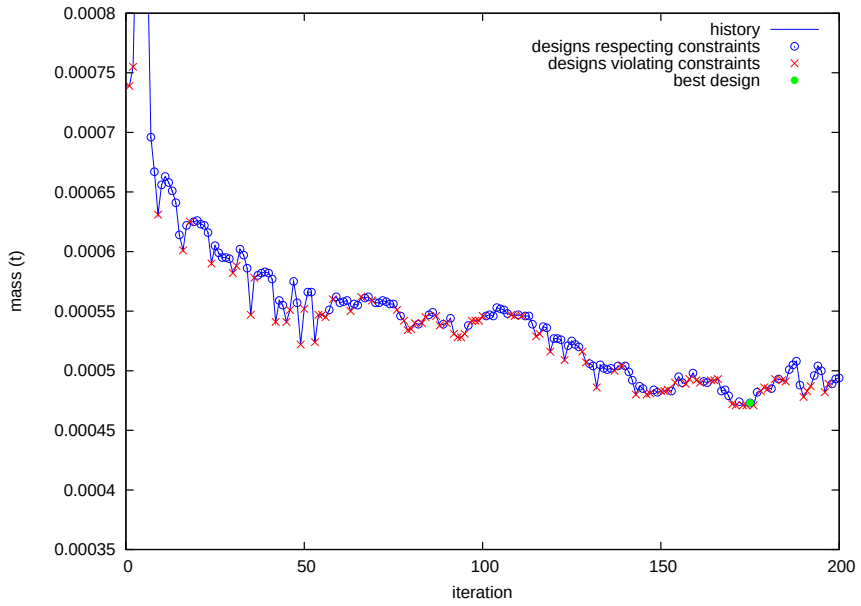


FIG. 6.27 – Three point bending, optimisation history for low velocity.

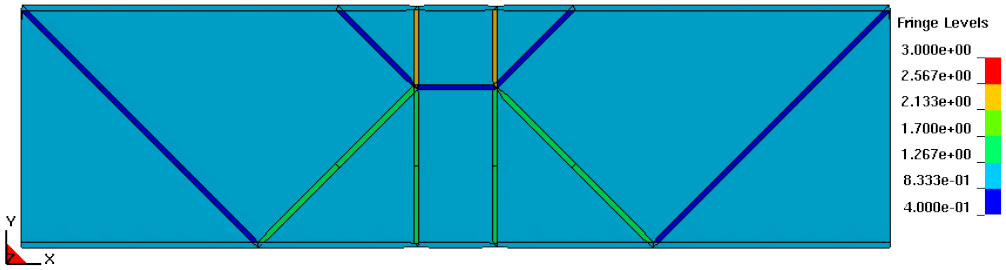


FIG. 6.28 – Three point bending, best design for intermediate velocity, thickness distribution (mm).

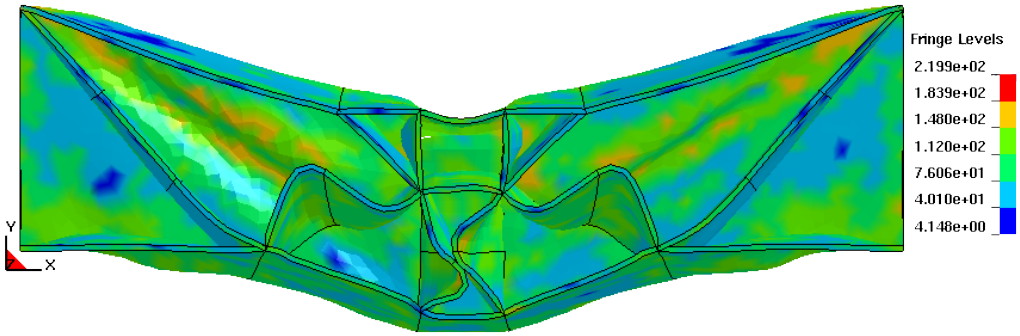


FIG. 6.29 – Three point bending, best design for intermediate velocity, von Mises stress under deformation (MPa).

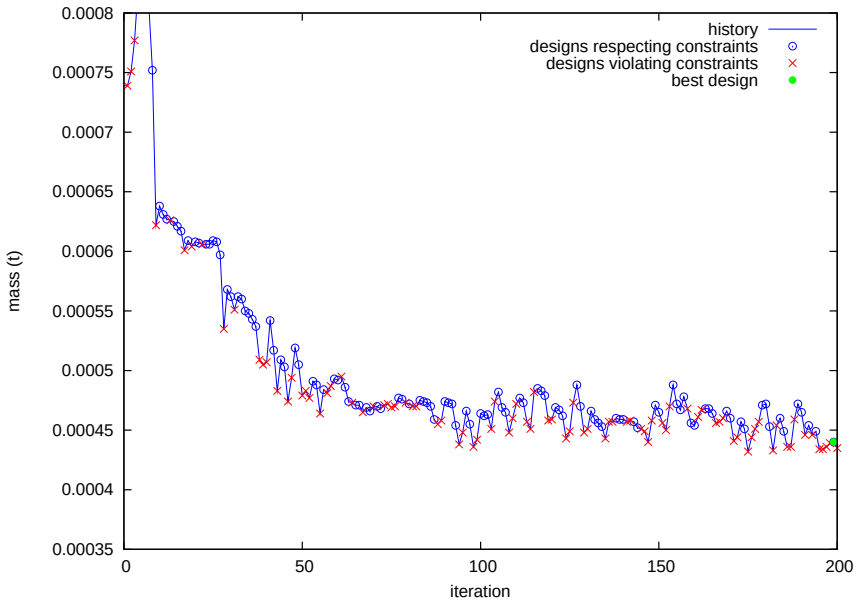


FIG. 6.30 – Three point bending, optimisation history for intermediate velocity.

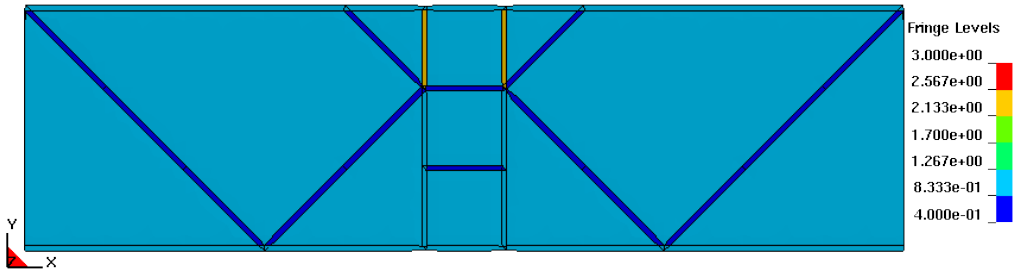


FIG. 6.31 – Three point bending, best design for high velocity, thickness distribution (mm).

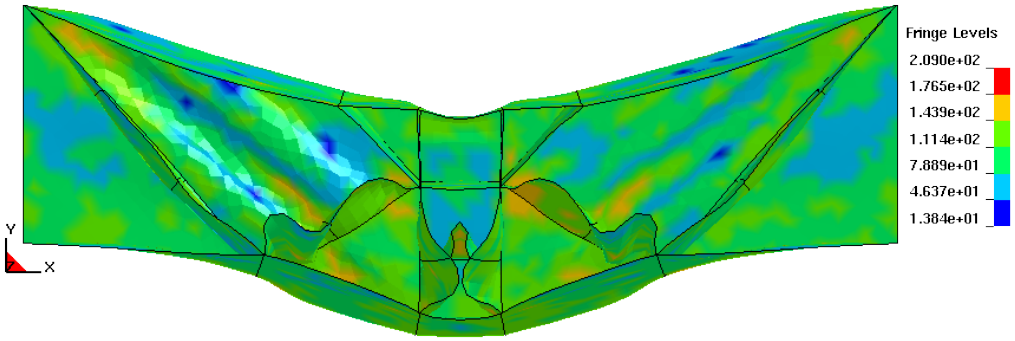


FIG. 6.32 – Three point bending, best design for high velocity, von Mises stress under deformation (MPa).

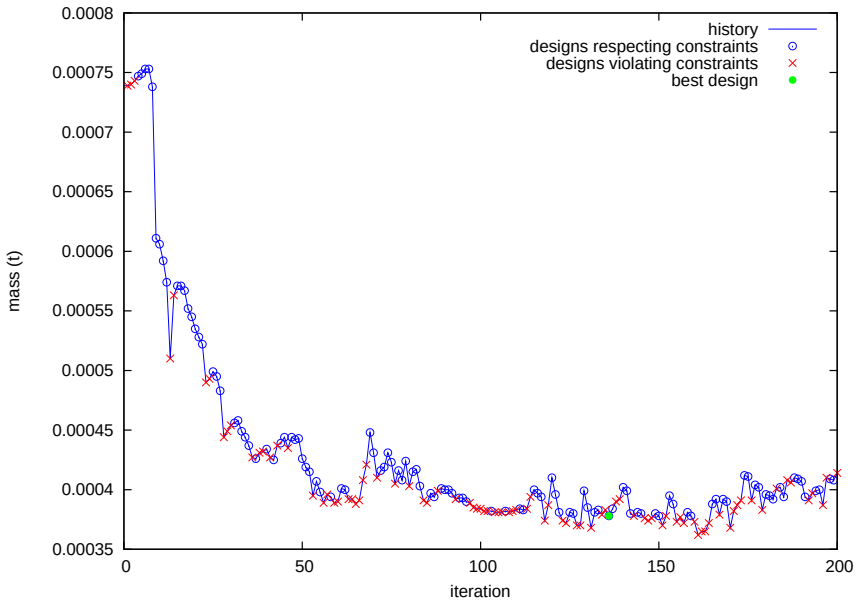


FIG. 6.33 – Three point bending, optimisation history for high velocity.

Chapter 7

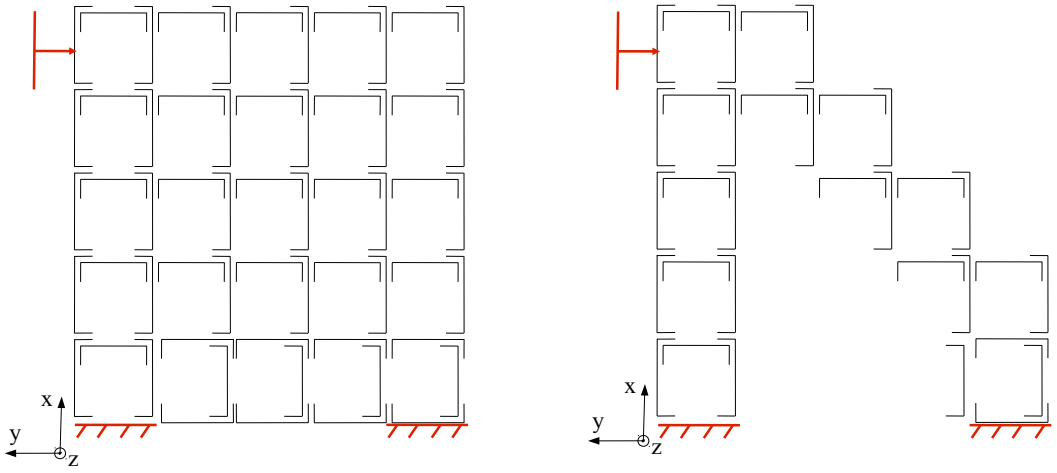
Industrial applications with HCATWS

The different application examples studied in Chapter 5 and Chapter 6 demonstrate the interest of the HCATWS method and provide confidence in its ability to derive optimised structures. However, it is critical to assess the method on more elaborate and realistic applications in order to identify possible bottlenecks and define how to improve the method further.
The application presented in Section 7.1 is a simplified side impact. While all applications presented in Chapter 5 and Chapter 6 deal with 2D design spaces, this application deals with a 3D design space. This simple difference allows to uncover many challenges not discussed so far. Another application example is presented in Section 7.2 where a car component is optimised within a full vehicle simulation context. In this application, the HCATWS method is therefore tested within a completely realistic design case.

7.1 First application: side impact

7.1.1 Problem definition

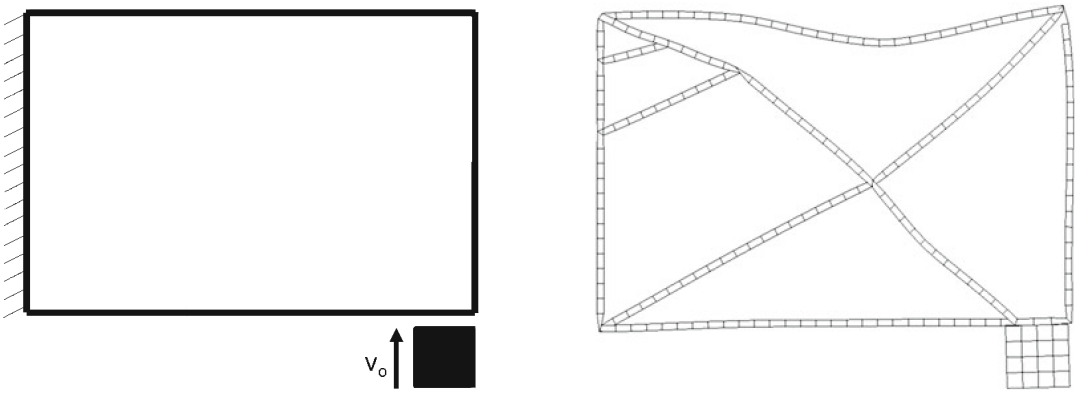
As mentioned in introduction, this test is the first HCATWS application using a 3D space filling. Therefore, the topologies derived cannot necessarily be extruded as it is the case in Chapter 5 and Chapter 6. A different manufacturing method should be used which raises the question of cells connectivity (see Section 7.1.2).
The design space is voluntarily simple being a cuboid of $1000 \times 1000 \times 100$ mm (see Figure 7.1(a)). The bottom left and right corners are constrained in all directions and a finite rigid wall moving in y-direction impacts the top-left corner. Figure 7.1(b) illustrates a possible topology which could be derived with the HCATWS method. A similar problem was studied by Ortmann *et al.* [94] (see Figure 7.2) with a fixed outer boundary to the design space.



(a) Design space and load case

(b) Possible topology

FIG. 7.1 – Side impact, problem definition.



(a) Design space and load case

(b) Best topology derived with minimum impactor displacement objective

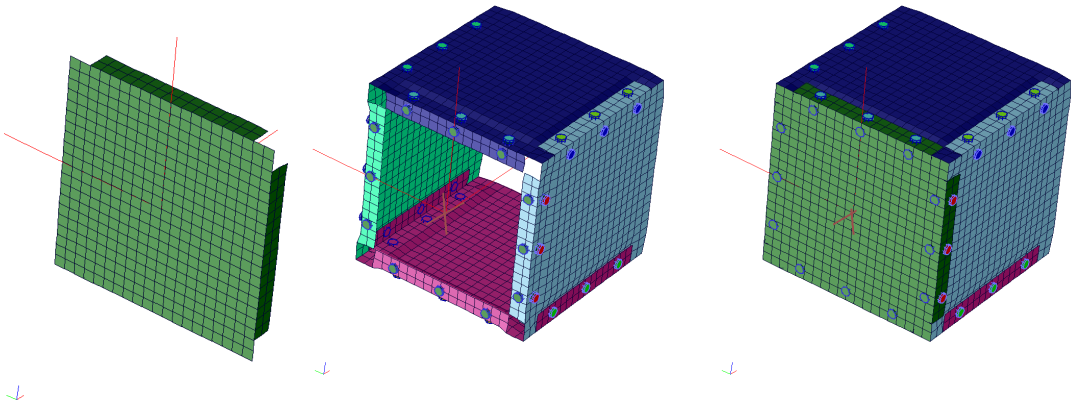
FIG. 7.2 – Similar side impact problem studied with Graph and heuristic based method (*Source: [94]*).

The cubic 3D space filling described in Section 4.2.2 is used here with 10 cubes in x- and y-directions and 1 cube in z-direction. Therefore, each cube of the space filling has a side of 100 mm. To avoid redundancy, the space filling is made of 10 "5 walls boxes" and 90 "4 walls boxes", for a total of 410 walls or 410 HCATWS cells. The structure is impacted by a rigid wall with a mass of 920 kg and an initial velocity of 10 m.s^{-1}. The objective is to minimise the mass of the structure while respecting a maximum displacement constraint of d_{max}=300 mm. The structure is made of a high strength steel modelled with a Johnson-Cook plasticity law [66]. The material parameters are: $\rho = 7.82\,10^{-9}$ t.mm^{-3}, E=2.1 10^5 MPa, $\nu = 0.3$, $a = 466$ MPa, $b = 805$ MPa, $n = 0.55$, $c = 0.11$, $\dot{\varepsilon}_0^p = 13.46$ mm.s^{-1}). Where a, b, c, n, $\dot{\varepsilon}_0^p$ are the Johnson-Cook plasticity parameters. The parameter values were given by an industrial partner.

7.1.2 Modelling and algorithm modifications

Cells connections

The previous section raised the question of cells connections. One of the main joining techniques in automotive body-in-whites is to spot weld different metal sheets. The cells connections in the model discussed here are defined to reflect this. As each cell of the space filling is independent and can be removed from the model at any point of the optimisation process, the connections need to be as flexible as possible. To this effect, the cells are made not only of a main square wall but also of four smaller connection areas, perpendicular to their main wall (see Figure 7.3(a)). These connections areas allow not only to connect the cells within a cubic box of the space filling (see Figure 7.3), but also to connect them with the other cells of the space filling. They should offer a more realistic modelling than edge-to-edge connections. Currently, the connection pattern between the cells is fixed and identical all over the structure. As a result, if during the HCA process some cells are deleted, their connections areas are deleted too. Depending on the configuration between the different cell boxes, it may result on some cells becoming loose. Adaptive connections could be implemented to treat this specific question but this is not discussed here.



(a) Single cell (b) Cell connections (c) Complete cubic box assembly

FIG. 7.3 – Connections within a thin-walled cubic box.

The natural way to use the connection areas is to use spot welds (but gluing could also be used for instance). The definition of these spot welds, and therefore the relationship between the cells of the HCA is really important.
The dimensions of the connection areas are strongly influenced by the dimensions of the boxes. They should be as small as possible when compared to the main plane surface so that the bulk of the energy absorption occur in the main surface. Yet, they should be big enough to support spot welds. The width of the connection areas is chosen as 10 mm in order for them to include two mesh elements of 5 mm. The length of the connection areas (along the edge of the cells) is chosen as 80 mm. Given the dimensions of the cubic boxes and the width of the connection area, it is

the biggest value possible without overlapping of the different connection areas.
The spot weld diameter is chosen as 5 mm upon recommendation of an industrial partner. Given the length of the connection area, 3 spot welds are used per area. More seems unrealistic given the length of the connection area (80 mm) and less would not provide the same quality of connection. With 3 spot welds over 80 mm, the distance between them is set to 27 mm.
Eventually, the mechanical behaviour of the spot welds is modelled as linear without failure with the following material parameters: $\rho = 3\,10^{-9}$ t.mm^{-3}, E=1.733 10^5 MPa and $\nu = 0.3$. Once again, these values were adapted from an industrial model.

Naming convention

In order to accommodate the more advanced connections between the cells, the naming convention introduced in Section 4.4.2 is extended. Each cubic box of the space filling is made of an extruded SFE CONCEPT base section which is closed in both ends by user defined surfaces. With the addition of the connection surfaces, the base section gets more complicated and the user-defined surfaces need to be divided in order to define the multi-flanges.
The naming convention is therefore extended to the base section segment names and to the user-defined surfaces names (see Figure 7.4). This allows an easier automatic update of the model.

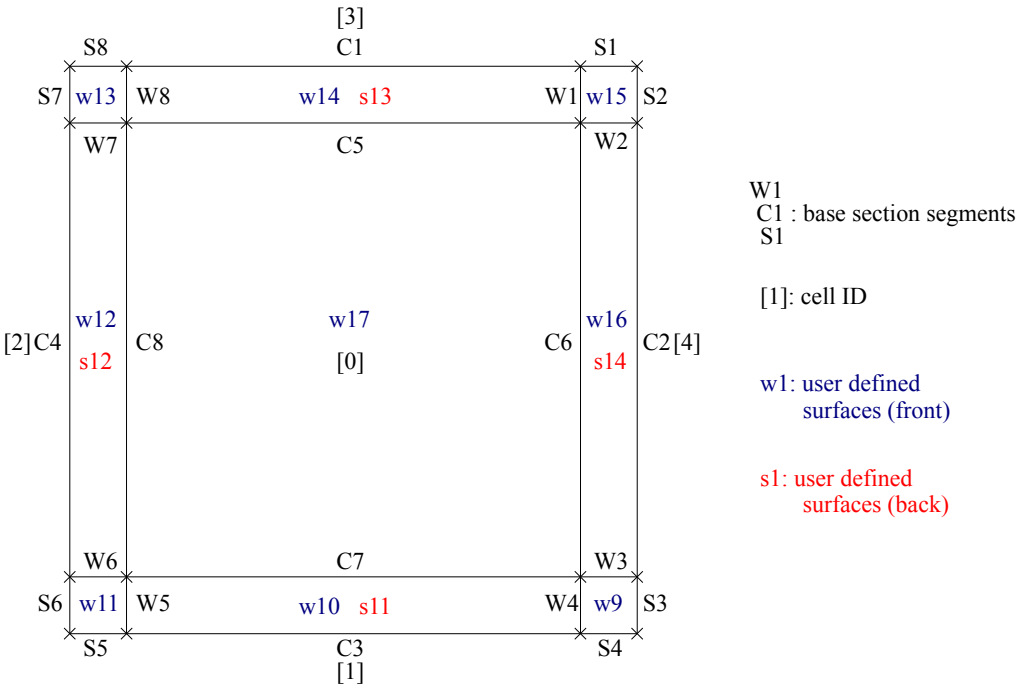


FIG. 7.4 – SFE CONCEPT base section and user defined surfaces.

Model update

As previously, the model update between iterations is done automatically with SFE CONCEPT. With the additional connection areas, special care need to be taken when a cell is deleted. As the connection area is shared with other cells, the area cannot simply be deleted from the model. The definition of the multi-flange itself need to be updated and reflect the new arrangement of cells it should include. Also, when the thickness value of a cell is modified, it has an effect on the multi-flanges it belongs to. The mesh of the model is modified by SFE CONCEPT taking this effect into account.

Algorithm

In order to minimise the influence of the spot weld parameters, the reaction forces going through the spot welds are monitored throughout the optimisation. A maximum spot weld reaction force level $\mathrm{F_{sw,max}}$ is set by the user and the algorithm is modified in order to make sure this level is not reached during the crash simulation.
The modification is implemented on the cells update rule. For each cell, the maximum reaction force value going through all its spot welds is identified. If this value is lower than $\mathrm{F_{sw,max}}$, the normal update rule is applied. If the value is higher, the update rule is bypassed and the thickness variation for this cell is directly set to δt_{max}.

7.1.3 Optimisation results

First run

A first optimisation is performed with the parameters given in Table 7.1. Given the load case, a symmetry constraint is applied in the Oxy direction. The maximum mass fraction change allowed $\Delta M_{f,\mathrm{max}}$ is set constant equal to 0.3. Similarly, the thickness increment parameter ζ is set constant equal to 0.5 mm.

t_{min}	t_{max}	k_{max}	ε_1	ε_2	δt_{max}
0.4 (mm)	3.0 (mm)	100	1e-3	0.5 (mm)	0.5 (mm)
l_0	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$	$b_{0,5}$
5	0.05	0.1	0.2	0.4	0.7
l_q	$b_{q,1}$	$b_{q,2}$	l_r	$b_{r,1}$	ζ
2	0.1	0.5	1	0.1	0.05

TAB. 7.1 – Parameters for first optimisation runs.

The reference structure with all the cells active and with a thickness of 1.0 mm has a mass of 41.234 kg and the maximum impactor displacement is 367.5 mm. For this optimisation run, the best design is found at iteration 28 (see Figure 7.5) with a mass of 30.568 kg (or 25.9 % less than the reference) and a maximum displacement of 294 mm. The last design (iteration 100, see Figure 7.6) has a higher mass (32.963 kg, or 20.1 % less than the reference) and similar displacement (291 mm) but its topology is more clearly defined with 161 out of 410 active cells instead of 199 at iteration 28.

The optimisation history is illustrated in Figure 7.7. Large oscillations can be observed in the first part of the optimisation and stagnation in the end of it. This would suggest that the parameters chosen here are not appropriate.

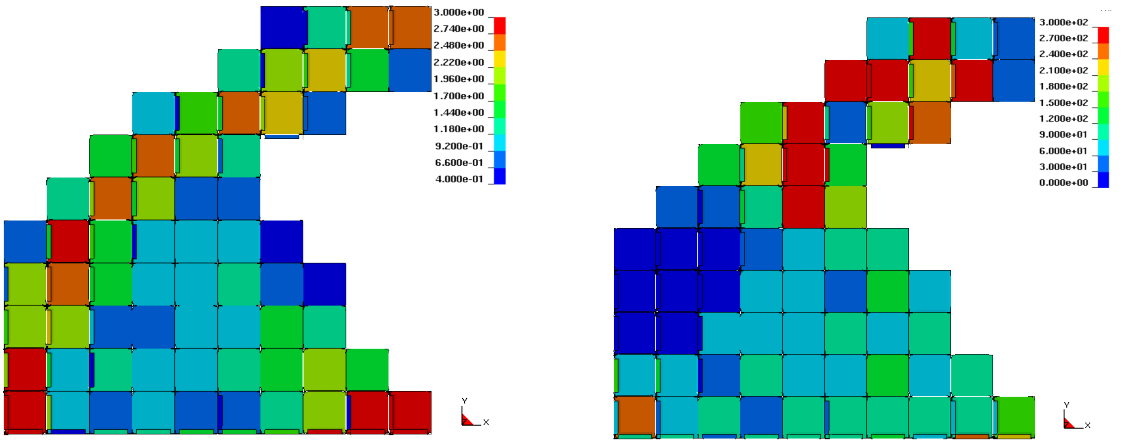


FIG. 7.5 – Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 28$^{\text{th}}$ iteration (initial state).

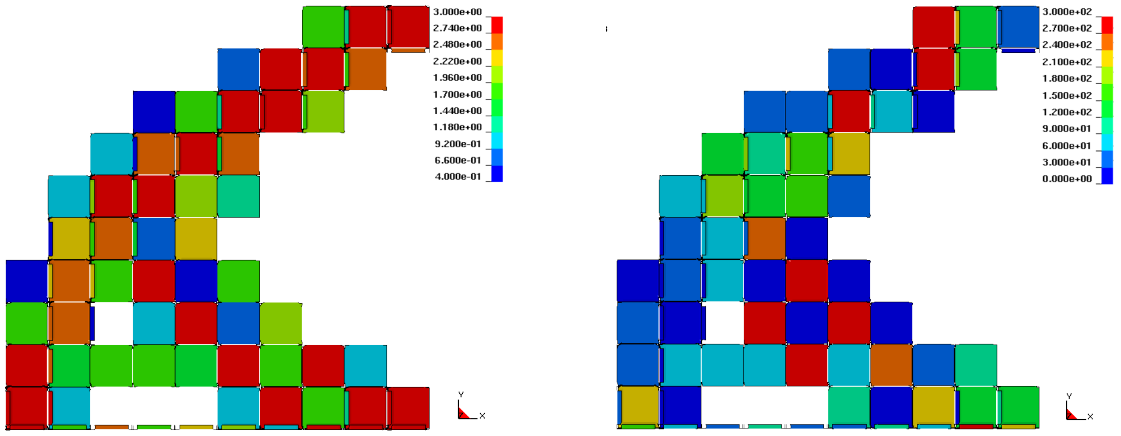


FIG. 7.6 – Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 100$^{\text{th}}$ iteration (initial state).

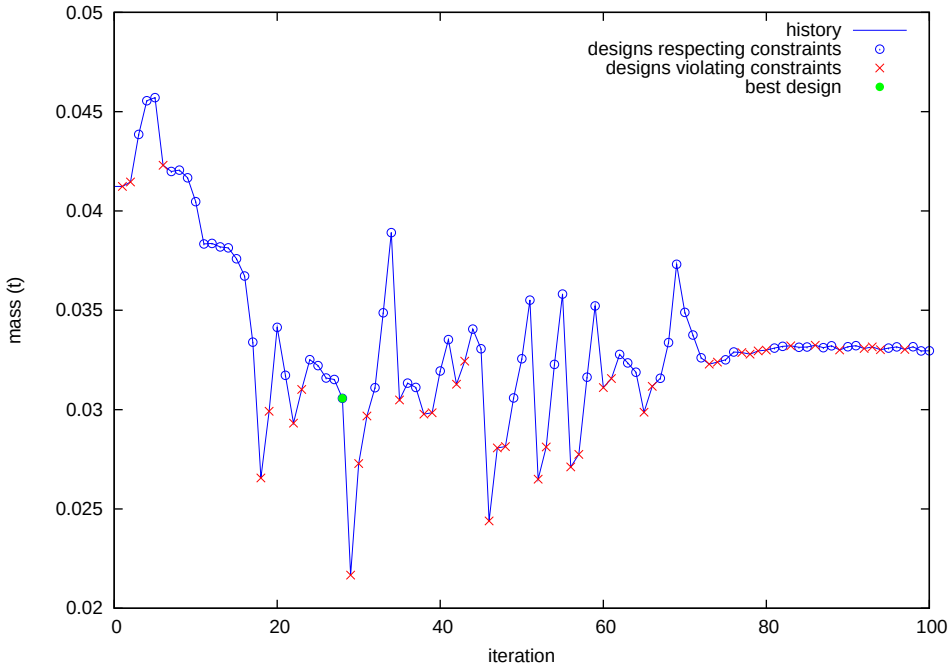


FIG. 7.7 – Optimisation history, mass value.

Second run

A second optimisation run is performed with more advanced settings. The maximum mass fraction change $\Delta M_{f,\max}$ is a linearly decreasing function as defined in Equation (4.5) where $\eta = 0.21$ and thickness increment parameter ζ is defined as per Equation (4.14) where $\theta = 3$. The other parameters remain the same as for the previous run.

The best design is found at the 80^{th} iteration with a mass of 28.055 kg (or 32.0 % less than the reference) and a maximum displacement of 297.2 mm. This is a significant improvement when compared to the previous run. On Figure 7.8 the Internal Energy Density distribution seems more homogeneous than for the previous runs. Nevertheless, the structure is still pretty complex with 306 out of 410 remaining active cells.

The optimisation history is given in Figure 7.9. The oscillations are smaller than for the previous run and a steadier progression of the optimisation can be observed.

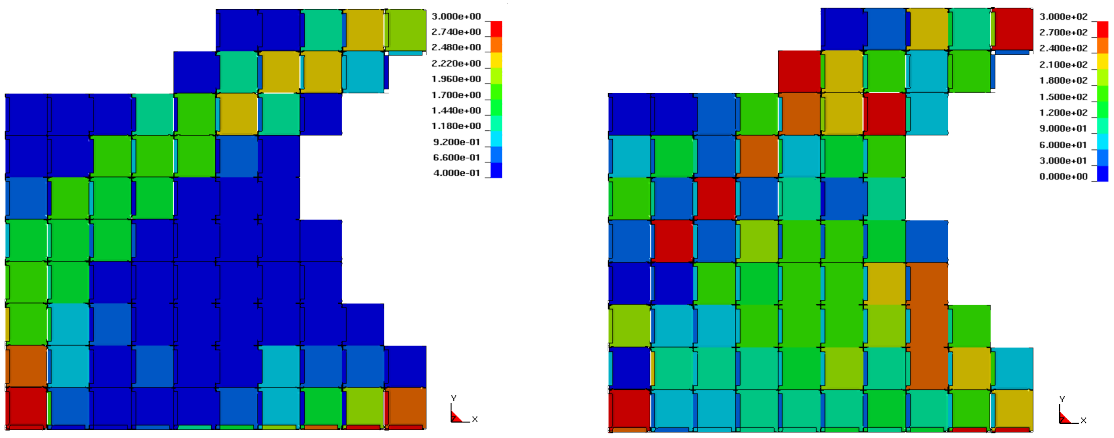


FIG. 7.8 – Thickness distribution (mm) (*left*) and IED (J.kg^{-1}) (*right*) at 80$^{\text{th}}$ iteration (undeformed state).

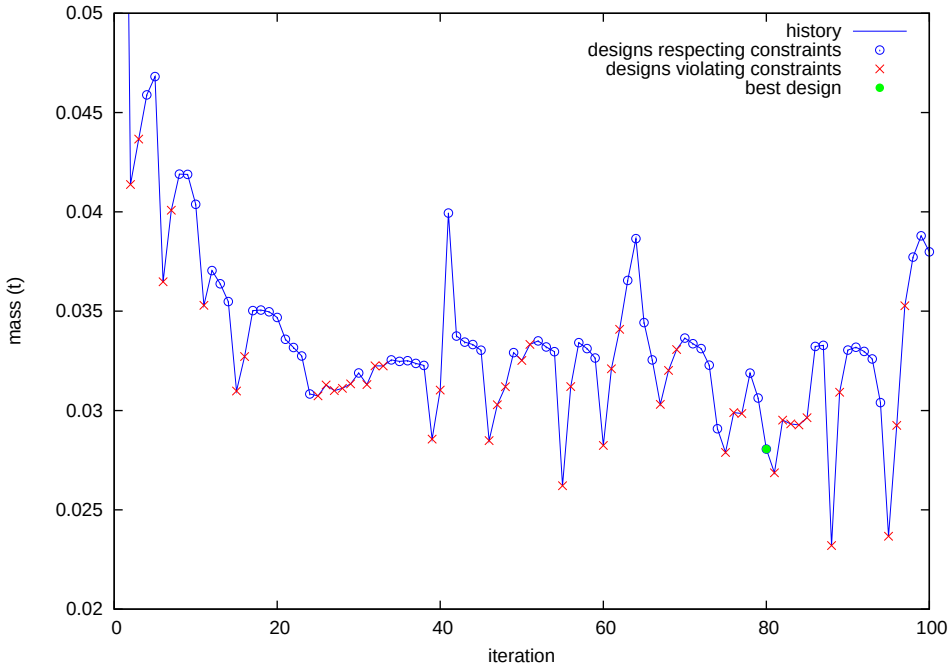


FIG. 7.9 – Optimisation history, mass value.

7.2 Second application: pole impact

7.2.1 Problem definition

This application example is taken from the same collaborative project as the problem detailed in Section 5.5.1 (see [9]). Once again it is a pole impact but this time, the simulation is performed for a full vehicle. The design task remains the optimisation of an extruded rocker's cross-section, but with a much more complex design space. The objective is to minimise the mass of the rocker while keeping the reduction of survival space due to the impact with the pole under 170 mm. Here, the survival space is defined as the distance between the central tunnel of the vehicle

and the rocker. Its measurement is aligned with the pole, where the intrusion is the highest. The reference rocker provided with the model has an initial mass of 15.217 kg and a survival space reduction of 175 mm.

7.2.2 Topology optimisation with square space filling

The geometry of the model being more complex than for the applications presented so far, the design space definition is not straightforward. Two design spaces are defined and filled using the square tiling described in Section 5.1.1 (see Figure 7.10). The first design space, or reduced design space, is bounded not only by the outer boundaries of the reference rocker, but also, by some inner features from the reference cross-section. This allows generating a design space close to a parallelogram with sides of 156 and 90 mm and a wider angle of 93°. Since this angle is close to 90°, it is decided to turn the base square of the space filling into a rhombus which follows the principal directions of the parallelogram. The length of the cells is chosen as 22 mm as it has multiples close to 90 and 156 ($22 \times 4 = 88$ and $22 \times 7 = 154$). The space filling is therefore made of 4 cells in the primary direction and 7 cells in the secondary direction, for a total or 45 cells. To accommodate the geometry, some small distortions are necessary near the boundaries.
The second design space, or extended design space, is bounded only by the outer boundaries of the reference rocker. The extended space filling uses the reduced space filling as a base and extends it to the new boundaries. Because of the geometry, more distortions are needed to complete the space filling (see Figure 7.10). In this second case, the number of cells is 71.

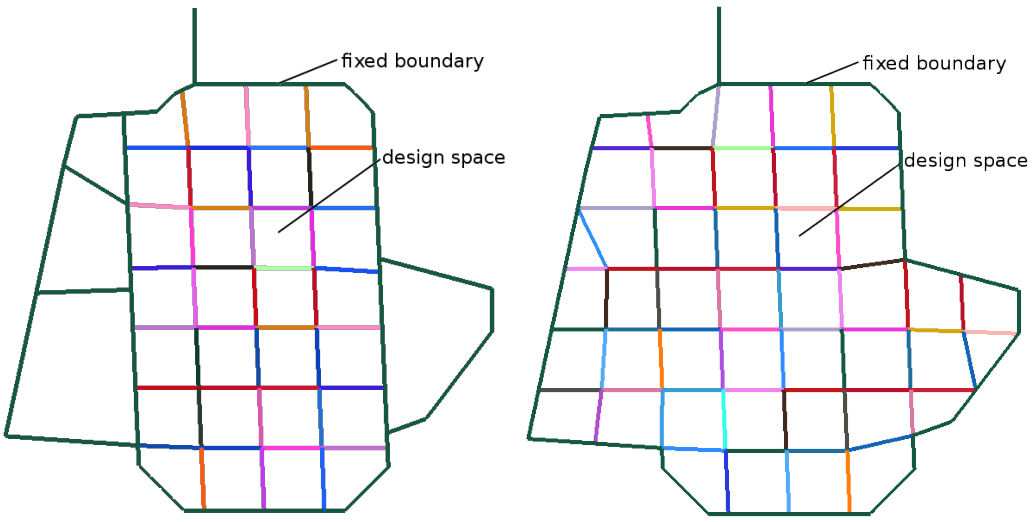


FIG. 7.10 – Full vehicle pole impact, square space fillings for reduced (*left*) and extended (*right*) design space.

Due to the expensive nature of this crash simulation, only a limited number of configurations were tested for optimisation. For both the reduced and the extended design space, the optimisation parameters are given in Table 7.2.

$t_{\min}$	$t_{\max}$	$k_{\max}$	ε_1	ε_2	$\delta t_{\max}$
0.4 (mm)	3.0 (mm)	100	1e-3	0.5 (mm)	0.5 (mm)
η	τ	θ	$M_{f,0}$		
0.4	10	3	0.5		
l_0	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$	$b_{0,5}$
5	0.05	0.1	0.2	0.4	0.7
l_q	$b_{q,1}$	$b_{q,2}$	l_r	$b_{r,1}$	
2	0.1	0.5	1	0.1	

TAB. 7.2 – Parameters for optimisation with square space filling.

τ is chosen voluntarily low in order to quickly decrease the mass fraction changes and therefore promote a quick convergence. Also, in order to help with convergence, the reintroduction of cells is not enforced. For the reduced space filling, the best design is found at iteration 70 with a mass of 14.969 kg (1.6% less than the reference structure) and an survival space reduction of 169.9 mm. 15 of the original 45 cells remain active (see Figure 7.11, *left*). For the extended space filling, the best design is found at iteration 100 with a mass of 13.372 kg (12.1% less than the reference structure) and an intrusion of 169.2 mm. 37 of the original 71 cells define the topology (see Figure 7.11, *right*). In both cases the mass has been reduced when compared to the reference structure and the constraint is now respected.

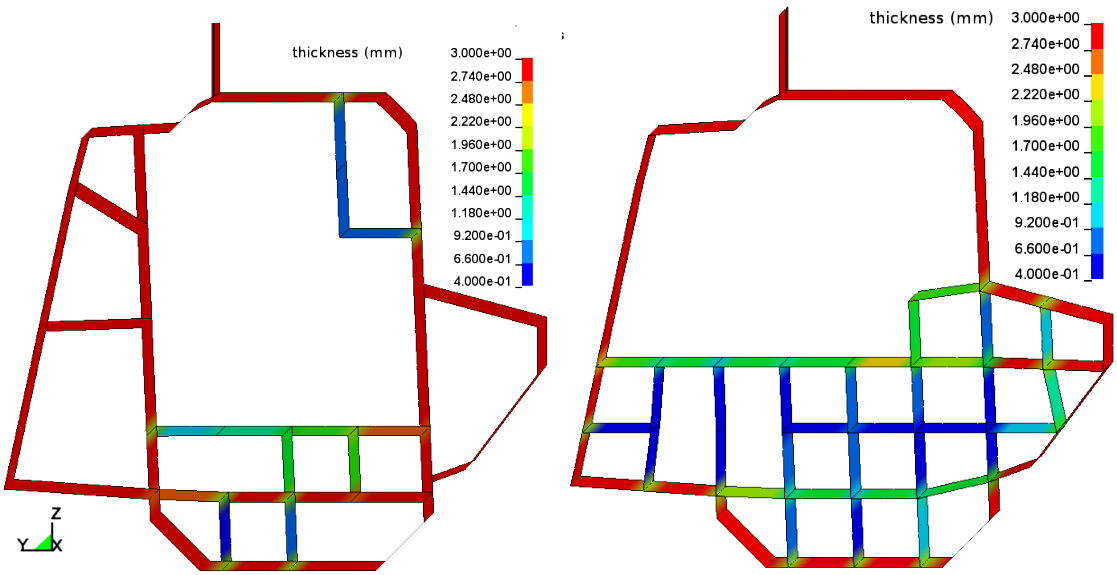


FIG. 7.11 – Full vehicle pole impact, best topology for reduced (*left*) and extended (*right*) square space fillings, thickness distribution (mm).

The Internal Energy Density distribution of both designs is given in Figure 7.12. In both cases, the IED level is in the same range for all cells. For the extended design space, the average IED level is higher than for the reduced design space. This suggests a better use of material in the extended design space which keeps only the boundaries of the reference structure. This statement is backed up by the

final deformations of both designs. For the reduced design space (see Figure 7.13), the reinforcements from the reference structure on the left part of the rocker do not deform much and therefore do not contribute much to the energy absorption. For the extended design space (see Figure 7.14), all the reinforcements are strongly deformed.

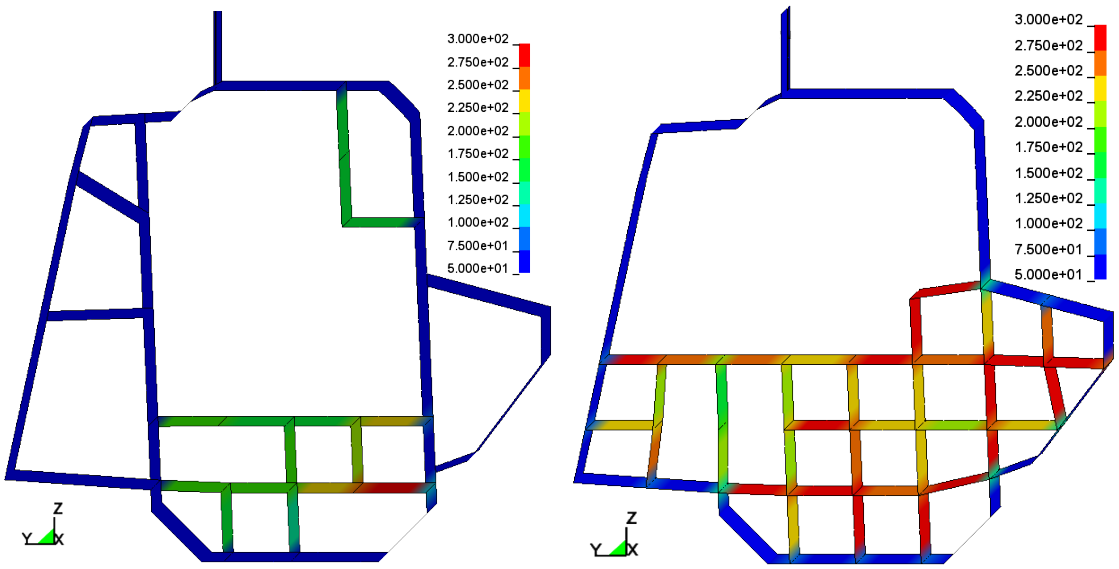


FIG. 7.12 – Full vehicle pole impact, IED distribution for reduced (*left*) and extended (*right*) square space fillings ($J.kg^{-1}$).

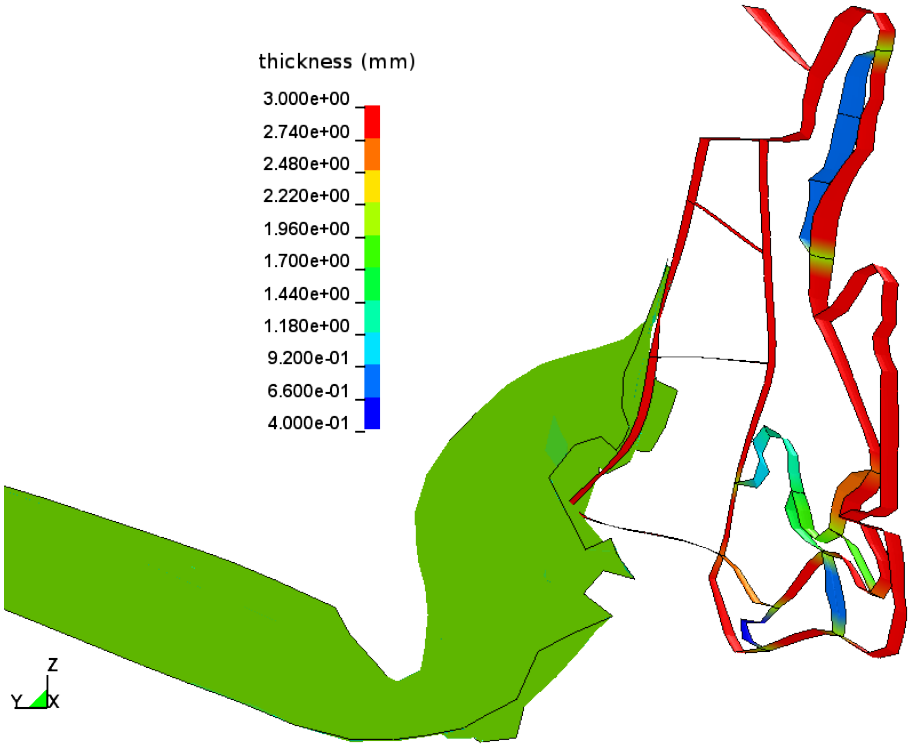


FIG. 7.13 – Full vehicle pole impact, deformation of best design for reduced square space filling, thickness distribution (mm).

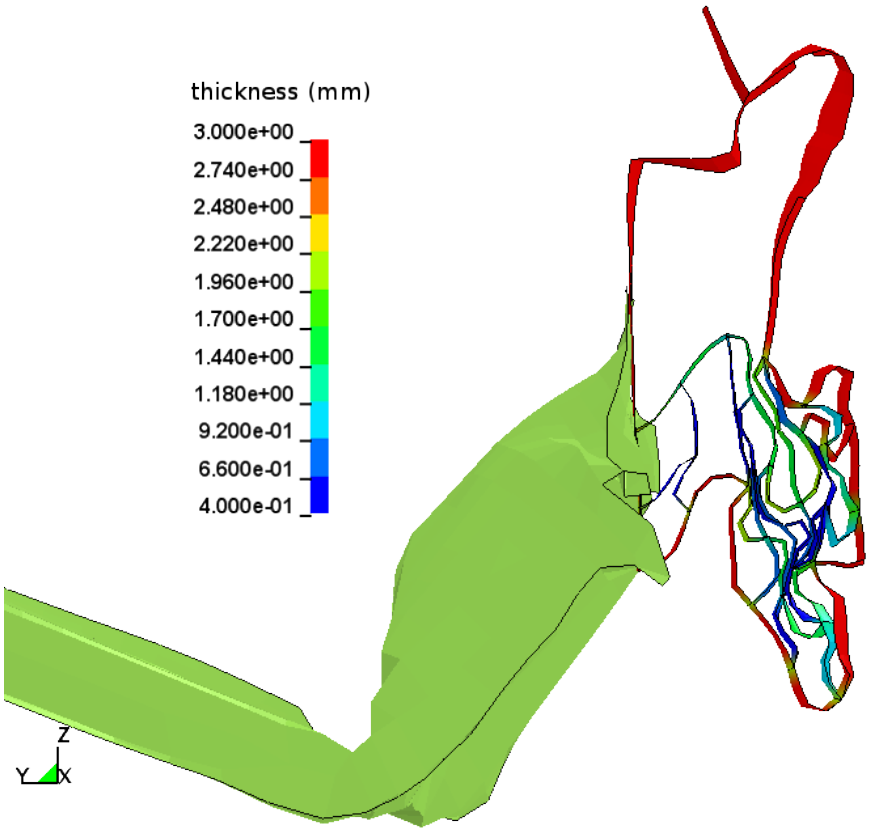


FIG. 7.14 – Full vehicle pole impact, deformation of best design for extended square space filling, thickness distribution (mm).

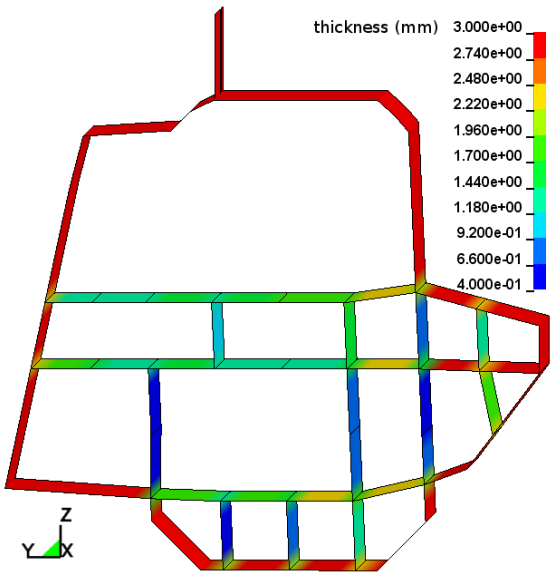


FIG. 7.15 – Full vehicle pole impact, alternative topology for extended square space filling, thickness distribution (mm).

An alternative optimisation run is performed for the extended space filling. Only the initial mass fraction is changed from 0.5 to 0.8. The best design is found at iteration 66 (see Figure 7.15) with a mass of 13.768 kg (9.5% less than the reference structure) and a survival space reduction of 169.0 mm. Although the mass is higher than previously, the topology is more clearly defined with 32 of the original 71 cells active.

7.2.3 Topology optimisation with diagonal enriched space filling

In order to try to find a better topology, the problem is also solved using the diagonal enriched space filling. Due to limited computational resources, only the reduced design space is studied here. The space filling is based on the square space filling described in the previous section. With the diagonal cells, it is made of 98 cells (see Figure 7.16). The optimisation parameters are similar as the ones used previously. In order to delete more cells in the beginning of the optimisation τ is now set to 30. The additional update rule parameters for the diagonal neighbours are defined as in Table 6.1.

The best design is found at iteration 83 with a mass of 14.640 kg (3.8% less than the reference structure and 1.4% less than with the reduced square space filling) and a survival space reduction of 168.9 mm. 16 out of the initial 98 cells define the topology, including 3 diagonal cells (see Figure 7.17). This is really similar to the 15 cells remaining with the square space filling. The topology is different here though, and the deformation of the cross-section (see Figure 7.18) seems to be more uniform in the z-direction than for the square space filling (see Figure 7.13).

In this case, the use of the diagonal enriched space filling gives a better result than the square space filling.

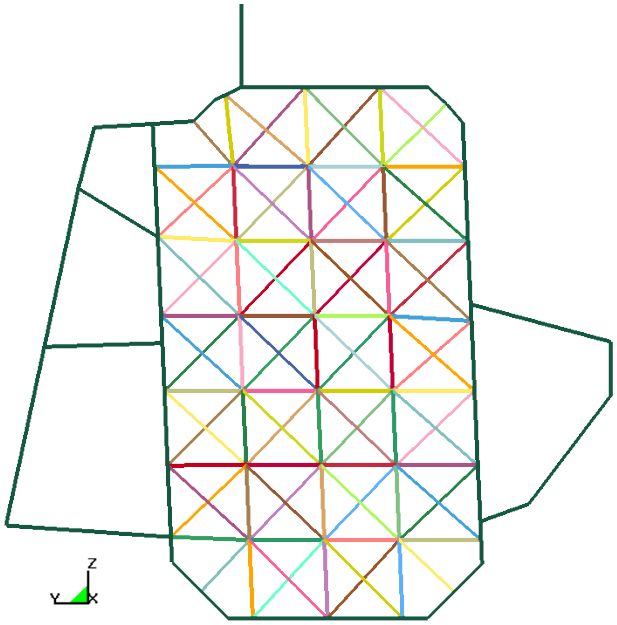


FIG. 7.16 – Full vehicle pole impact, diagonal enriched space filling for reduced design space.

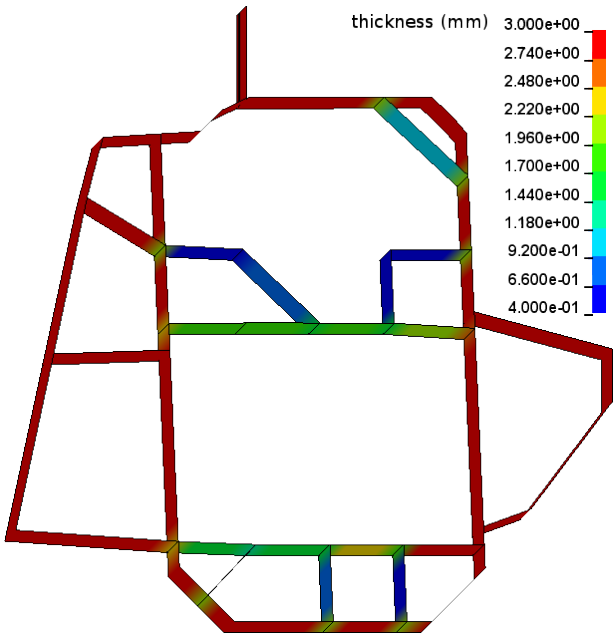


FIG. 7.17 – Full vehicle pole impact, best topology for reduced diagonal space filling, thickness distribution (mm).

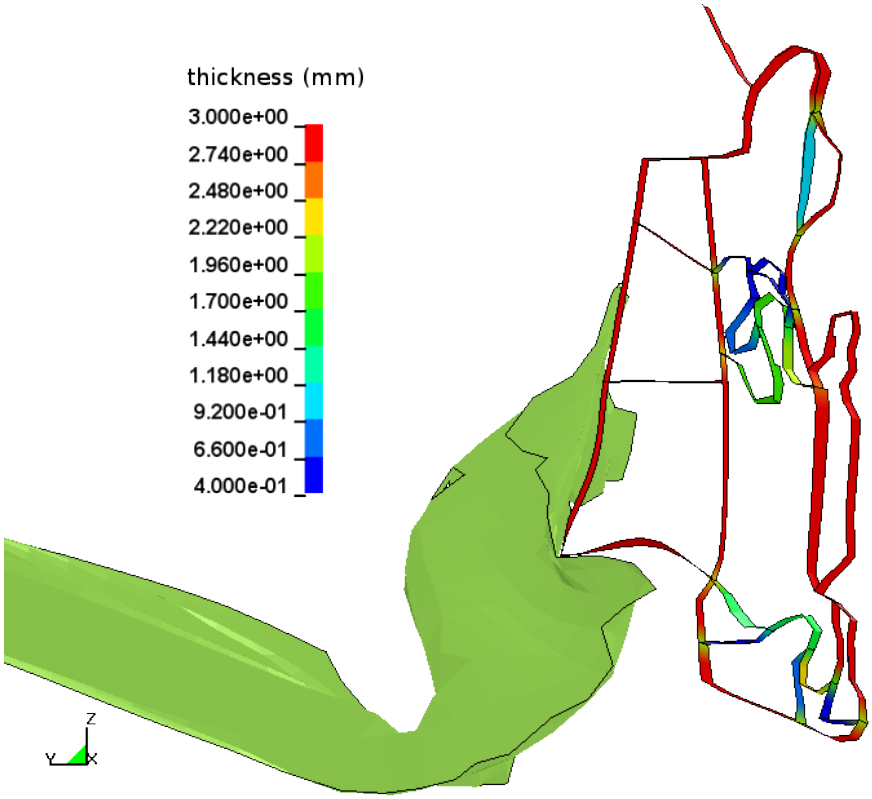


FIG. 7.18 – Full vehicle pole impact, deformation of best design for reduced diagonal space filling, thickness distribution (mm).

Chapter 8

Conclusion and future work

8.1 Achievements

8.1.1 Topology optimisation framework

First and foremost, the main achievement of this research project has been the establishment of a topology optimisation framework for crashworthiness design. This framework is based around the geometry, topology and finite element model update capabilities of the CAE software SFE CONCEPT [109] (as described in Section 4.4) while the HCATWS algorithm is implemented in Scilab [108]. This algorithm is inspired both by the HCA and by the Ground Structure Approach applied to thin-walled structures and was developed after investigating the different methods already available for topology optimisation in crashworthiness design.
The optimisation environment was tested on several academic and industrial problems. It was also tested in different environments, either on workstations or on a cluster without major difficulties.

8.1.2 Design space and space filling

The optimisation method is designed for different types of design space and offers different space filling methods. Indeed, the HCATWS method is presented for both 3D (Chapter 4 and Chapter 7) and 2D design spaces (Chapter 5, Chapter 6 and Chapter 7). For the 2D design cases, two different space fillings with a different level of complexity are investigated. Depending on the optimisation problem, the simple square space filling can perform better than the diagonal enriched space filling (see Section 5.3 and Section 6.3 for instance) or vice versa (see Section 5.2 and Section 6.2 for instance). For the 3D space filling, the investigations are only at their early stage and only one space filling is presented.
The size of the cells chosen for the space filling also has some influence on the results (see Section 5.2.2). In most cases though, the size of the cells is restricted by

the design space, by mechanical considerations and by manufacturing constraints. Overall, the definition of the space filling does have an influence on the optimisation results derived by the HCATWS method. This is a clear restriction, but this should be put in perspective with the fact that the method allows to derive feasible topologies within a reasonable amount of time which is not necessarily true with other methods.

8.1.3 Topology optimisation for static load cases

Although the HCATWS method is initially developed for crashworthiness design, its application to static load cases would be really advantageous. It would provide a single method for different applications, which is desirable in industrial environments. Also, this would allow to perform multidisciplinary optimisation with the HCATWS method, including both crash and static load cases.
Only one static problem is assessed in this thesis (see Section 5.2 and Section 6.2) as the priority was to evaluate the method for crashworthiness design. More investigation and comparison with other methods would be needed for validation, but the preliminary results are already encouraging.

8.1.4 Topology optimisation for crashworthiness design

As mentioned previously, the HCATWS method was developed specifically for crashworthiness design applications. This is reflected in the range of different crash load cases investigated. It includes a fully axial load case (see Section 5.3 and Section 6.3), an oblique load case (see Section 5.4 and Section 6.4) and different side cases (see Section 5.5, Section 6.5, Section 6.6, Section 7.1 and Section 7.2). Most of the applications address the optimisation of components for sub-system load cases. The application presented in Section 7.2 includes the component within the full vehicle for crash simulation while the application presented in Section 7.1 demonstrates the optimisation of a whole area of the vehicle.
The method already gives satisfying results for 2D design spaces. The axial load case is particularly interesting (see Section 5.3 and Section 6.3) with a derived topology matching the results found in literature. The three point bending case presented in Section 6.6 is also interesting as it showcases the influence of the transient nature of the crash on the derived topology. This puts into perspective the use of methods such as the ESLM to derive crashworthy structures. Eventually, it should be noted that as it stands, the homogeneous IED level target implemented in the algorithm seems to offer a better behaviour when the cells are all involved in the crash at the same time (as in the axial load case). More advanced IED targets could be implemented for other load cases, as suggested in the next section.

8.2 Further research

8.2.1 Implementation

The implementation and development of the method is still a work in progress and different ways to improve the method are considered. The different applications presented in this thesis show that the process for reintroduction of cells is a critical question. High reintroduction rate leads to oscillations of the algorithm and makes it more difficult to derive a properly defined topology. On the opposite, no reintroduction of cells may lead to deleting important cells with no possibility to reintroduce them in the design. In this regards, the reintroduction method detailed in Section 6.2.2 seems to be a good compromise. Yet, more advanced settings could be investigated here. The reintroduction could be defined to occur only if the design constraint is violated several times in a row. Alternatively, it could be defined to occur only in the early stages of the optimisation and be deactivated later in order to derive a well-defined topology.

Another area to investigate would be the homogeneous IED paradigm. For design constraints on force and accelerations or to have a better control on the kinematics of the crash (see for instance the oblique impact in Section 5.4, where the displacement is only controlled in one direction), a more advanced design target would be interesting. Different options could be implemented. Different IED targets could be defined for different regions of the design space. The different regions or targets could also be updated adaptively during the optimisation. Also, the IED level could be acquired at different times in the simulation for each region for a better control on the kinematics of the crash. Another option here would be to try and implement local heuristics on the cells, in a similar fashion as in the Graph and heuristic based method (see [94]).

For the space filling, variants could be implemented such as the triangular (see Figure 5.3), triangular-hexagonal or rhombille tilings (see Figure 5.4) for 2D design spaces. For specific application or manufacturing process, they could be an interesting alternative to the square space-filling. Also, for 3D design spaces, the square space filling could be enriched with diagonals, in a similar manner as for the 2D space filling. In this case, there would be six intersecting diagonal cells within each cube of the space filling.

In terms of usage and problem definition, more work should be done towards the automation of the space filling definition. Using the capabilities of SFE CONCEPT, it would be ideal for the user to only define the design space, the directions of space filling and the number of cells in each direction to automatically generate the space filling.

8.2.2 Applications

After testing the HCATWS method mostly on small applications examples, it is crucial to investigate more complicated and realistic examples such as the ones presented in Chapter 7. Such problems would enable to validate the method for production. Ideally, the method should be tested for problems such as deriving the topology for a full front-end of a car with realistic packaging constraints. This would be particularly interesting in the case of a vehicle without a front-engine, where the packaging constraint would be less restrictive. Also, more work should be done on applications with 3D design spaces. The use of edge-to-edge connections in this case should be compared to the spot weld connections proposed in Section 7.1.1.
Another important type of application for the HCATWS would be multidisciplinary optimisation problems. This would need first to be implemented in the algorithm. The coupling of different load cases should not be a problem as it has already been defined for the classic HCA (see [79]). In this case, the multi-objective (or multi constraint) problem is transformed into a single objective problem by weighing the outputs of the different load cases. Given the cost of the crash simulations, this seems more reasonable than building a Pareto frontier. Yet, it could be interesting to build some sort of Pareto frontier using the HCATWS method, maybe using some dynamic weights for the outputs.

8.2.3 Post-treatment of the results

For an easy transition to practical design, more work should be done on the post-treatment of the results from the HCATWS method. In Section 5.5, an example is given where shape optimisation is carried on the topology derived with the HCATWS method. In this case, the definition of the shape optimisation problem is done manually. Yet, with the capabilities of the software SFE CONCEPT, it would be possible and convenient to automatically set up the shape optimisation problem.
A question arising with such sequential optimisation is the definition of objective and constraint for the topology optimisation. Indeed, the result of the topology optimisation process is not the final design and therefore, should not necessarily satisfy the same constraints as the result of the shape optimisation. It is therefore possible to over-constrain or under-constrain the topology optimisation, and then set the proper objective and constraint for the shape optimisation. The topology optimisation can also simply be stopped when a pre-defined number of cells is deleted, regardless of whether the design can be improved further or not. Clearly, this approach offers some level of flexibility. It would be interesting to investigate it and possibly quantify it.
Another aspect of the results post-treatment is the definition of a clear topology. Sometimes the results are not necessarily a well defined topology (see for instance

Figure 7.11, *right*). In such cases, the result need to be interpreted to generate a clear structure and the performance of the interpreted structure is not necessarily the same. It would be better to be able to implement manufacturing constraints to define the topology more clearly, during the optimisation process.

8.3 Publications

The research detailed in this thesis has been presented by the author in the following academic and industrial conferences:

S. Hunkeler, F. Duddeck, M. Rayamajhi, and H. Zimmer. Pseudo topology optimisation for crashworthiness design, Hybrid Cellular Automata adapted to thin-walled structures. In *4th GACM Colloquium on Computational Mechanics*, Dresden, Germany, 2011.

S. Hunkeler, F. Duddeck, and M. Rayamajhi. Topology Optimisation Method for Crashworthiness design using Hybrid Cellular Automata. In *10th World Congress on Computational Mechanics*, São Paulo, Brazil, 2012.

S. Hunkeler, F. Duddeck, and M. Rayamajhi. Topology optimisation method for crashworthiness design using Hybrid Cellular Automata and thin-walled ground structures. In *9th European LS-DYNA Users' Conference*, Manchester, UK, 2013.

For the last presentation, a full paper has also been submitted to the conference proceedings. It is available at `www.dynalook.com/9th-european-ls-dyna-conference`.

Bibliography

[1] N. Aage, M. Nobel-Jørgensen, C.S. Andreasen, and O. Sigmund. Interactive topology optimisation on hand-held devices. *Structural and Multidisciplinary Optimization*, 47(1):1–6, 2013.

[2] M.M. Abdalla and Z. Gurdal. Structural design using cellular automata for eigenvalue problems. *Structural and Multidisciplinary Optimization*, 26:200–208, 2004.

[3] W. Abramowicz. Thin-walled structures as impact energy absorbers. *Thin-Walled Structures*, 41:91–107, 2003.

[4] A. Alavia Nia and M. Parsapour. An investigation on the energy absorption characteristics of multi-cell square tubes. *Thin-Walled Structures*, 68:26–34, 2013.

[5] J.M. Alexander. An approximate analysis of the collapse of thin cylindrical shells under axial loading. *Quarterly Journal of Mechanics and Applied Mathematics*, 13(1):10–15, 1960.

[6] Altair Engineering, Troy, MI, USA. *OptiStruct 12.0 User's Guide*, 2013.

[7] J. Ambrosio. *Crashworthiness: Energy Management and Occupant Protection*, chapter Rigid multibody systems: the plastic hinge approach, pages 205–237. Springer, Vienna, 2001.

[8] N. Andelic. Torsional analysis of open section thin-walled beams. *University of Belgrade, Faculty of Mechanical Engineering Transactions*, 40:93–98, 2012.

[9] ASCS (Automotive Simulation Center Stuttgart), Dynamore GmbH, HAW Hamburg, SFE GmbH. Research project: Methodical and software-technical implementation of topology optimization for crash-stressed vehicle structures, 2009-2012.

[10] N. Aulig and I. Lepenies. A topology optimization interface for LS-DYNA. In *11th LS-DYNA German User's Forum*, Ulm, Germany, 2012.

[11] H. Van der Auweraer, T. Van Langenhove, M. Brughmans, I. Bosmans, N. El Masri, and S. Donders. Application of mesh morphing technology in the concept of vehicle development. *International Journal of Vehicle Design*, 43:281–305, 2007.

[12] T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1995.

[13] T. Belytschko, W.K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures.* Wiley, Chichester, 2000.

[14] M.P. Bendsøe. Optimal shape design as a material distribution problem. *Structural Optimization*, 1:193–202, 1989.

[15] M.P. Bendsøe, A. Díaz, and N. Kikuchi. Topology and generalized layout optimization of elastic structures. In M. Bendsøe and C. Mota Soares, editors, *Topology design of structures*, pages 159–206. Kluwer Academic Publishers, 1993.

[16] M.P. Bendsøe and O. Sigmund. *Topology Optimization; Theory Methods and Applications.* Springer, Berlin, 2003.

[17] C. Bisagni. Crashworthiness of helicopter subfloor structures. *International Journal of Impact Engineering*, 27(10):1067–1082, 2002.

[18] B. Bochenek and K. Tajs-Zielińska. Local rules of cellular automata for generating optimal topologies in structural design. In *ECCM 2010, IV European Conference on Computational Mechanics*, Paris, France, 2010.

[19] A. Boresi and R. Schmidt. *Advanced Mechanics of Materials, 6th Edition.* Wiley, 2003.

[20] H. Bremermann. Chemotaxis and optimization. *Journal of the Franklin Institute*, 297:397–404, 1974.

[21] D.S. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.

[22] CARHS GmbH. Safety companion. Technical report, Alzenau, Germany, 2012.

[23] CARHS GmbH. Safety companion. Technical report, Alzenau, Germany, 2013.

[24] E. Carter, S. Ebdon, and C. Neal-Sturgess. Optimization of passenger car design for the mitigation of pedestrian head injury using a genetic algorithm. In *GECCO 2005: Conference on Genetic and Evolutionary Computation*, Washington, USA, 2005.

[25] M. Cavazzuti, A. Baldini, E. Bertocchi, D. Costi, E. Torricelli, and P. Moruzzi. High performance automotive chassis design: a topology optimization based approach. *Structural and Multidisciplinary Optimization*, 44:45–56, 2011.

[26] R. Chiong. *Nature-inspired algorithms for optimisation.* Springer, Berlin, Germany, 2008.

[27] W.S. Choi and G.J. Park. Structural optimization using equivalent static loads at all time intervals. *Computer Methods in Applied Mechanics and Engineering*, 191:2077–2094, 2002.

[28] J. Christensen, C. Bastien, and M.V. Blundell. Effects of roof crush loading scenario upon body in white using topology optimisation. *International Journal of Crashworthiness*, 12(1):29–38, 2012.

[29] C.H. Chuang and R.J. Yang. Benchmark of topology optimization methods for crashworthiness design. In *12th International LS-DYNA Users' Conference*, Detroit, USA, 2012.

[30] R. Courant, K.O. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der Mathematischen Physik. *Mathematische Annalen*, 100(32), 1928.

[31] H.S.M. Coxeter. *Regular Polytopes*, chapter Tesselations and honeycombs, pages 58–74. Methuen, London, UK, 1948.

[32] T. Dietrich. *Neue virtuelle Methoden zur Integration einer numerischen Fussgängerschutz-Optimierung in den Entwicklungsprozess einer Motorhaube.* PhD thesis, Technische Universität München, Germany, 2013.

[33] M. Dorigo and T. Stützle. *Ant Colony Optimization.* MIT Press, 2004.

[34] H. Drucker, C.J.C. Burges, L. Kaufman, A.J. Smola, and V. Vapnik. Support vector regression machines. In *NIPS*, pages 151–161, 1996.

[35] F. Duddeck. Multidisciplinary optimization of car bodies. *Structural and Multidisciplinary Optimization*, 35(4):375–389, 2008.

[36] F. Duddeck, S. Hunkeler, H. Zimmer, L. Rota, and M. Zarroug. Evaluation of shape optimisation strategies for crashworthiness. In *Weimarer Optimization and Stochastics Days 6*, Weimar, Germany, 2009.

[37] F. Duddeck and K. Volz. A new topology optimization approach for crashworthiness of passenger vehicles based on physically defined equivalent static loads. In *International Crashworthiness Conference*, Milano, Italy, 2012.

[38] Dynardo GmbH, Weimar, Germany. *OptiSLang Ver 3.1.0, Documentation*, 2009.

[39] A. Erhart, P. Schumacher, N. Lazarov, and H. Müllerschön. Topology optimization with LS-TaSC and Genesis/ESL for crash-loading. In *11th LS-DYNA German User's Forum*, Ulm, Germany, 2012.

[40] Euro NCAP. Testing of electric vehicles (version 1.0), 2010.

[41] Euro NCAP. Assessment protocol - Overall Rating (Version 6), 2012.

[42] Euro NCAP. Pedestrian, Testing Protocol (Version 6.2), 2012.

[43] European Aluminium Association. Aluminium Automotive Manual. Technical report, Brussels, Belgium, 2002.

[44] European Parliament. Directive 96/79/EC of the European Parliament and of the Council, 1996.

[45] European Parliament. Directive 2002/24/EC of the European Parliament and of the Council, 2002.

[46] A. Florian. An efficient sampling scheme: Latin hypercube sampling. *Probabilistic Engineering Mechanics*, 279:123–130, 1992.

[47] A. Forrester, A. Sóbester, and A. Keane. *Engineering Design via Surrogate Modelling. A Practical Guide.* Wiley, 2008.

[48] J. Forsberg and L. Nilsson. Topology optimization in crashworthiness design. *Structural and Multidisciplinary Optimization*, 33:1–12, 2007.

[49] H. Fredricson. *Topology optimization of Vehicle body structures.* PhD thesis, Linköping University, Sweden, 2004.

[50] M. Gambling. Efficient Topology, Topometry and Sizing Optimisation for LS-DYNA Analysis Problems - Coupling LS-DYNA to VR&D GENESIS. In *LS-DYNA Anwenderforum*, Bamberg, Germany, 2008.

[51] Z. Geem, J. Kim, and G. Loganathan. A new heuristic optimization algorithm: harmony search. *Simulation*, 76(1):60–68, 2001.

[52] F. Glover. Heuristic for integer programming using surrogate constraints. *Decision Sciences*, 8(1):156–166, 1977.

[53] L. Guo, A. Tovar, C. Penninger, and J.E. Renaud. Strain based topology optimization for crashworthiness using hybrid cellular automata. *International Journal of Crashworthiness*, 16(3):239–252, 2011.

[54] P. Gupta. A study on inversion of metallic thin-walled conical shells. *International Journal of Crashworthiness*, 16(6):607–619, 2011.

[55] J. Halgrin, G. Haugou, E. Markiewicz, and L. Rota. Integrated simplified crash modelling approach dedicated to pre-design stage: evaluation on a front car part. *International Journal of Vehicle Safety*, 3(1):91–115, 2008.

[56] J. Hilmann. *On the development of a process chain for structural optimization in vehicle passive safety.* PhD thesis, Technische Universität Berlin, 2009.

[57] S. Hou, Q. Li, S. Long, X. Yang, and W. Li. Multiobjective optimization of multi-cell sections for the crashworthiness design. *International Journal of impact engineering*, 35(11):1355–1367, 2008.

[58] X. Huang, Y. Xie, and G. Lu. Topology optimization of energy-absorbing structures. *International Journal of Crashworthiness*, 12(6):663–675, 2007.

[59] S. Hunkeler, F. Duddeck, M. Rayamajhi, and H. Zimmer. Shape optimisation for crashworthiness followed by a robustness analysis with respect to shape variables - example of a front rail. *Structural and Multidisciplinary Optimization*, 48(2):367–378, 2013.

[60] Impact Design Europe, Michalowice, Poland. *Visual Crash Studio v2.2 User's Manual*, 2012.

[61] N. Inou, N. Shimotai, and T. Uesugi. A cellular automaton generating topological structures. In *Proceedings of the 2nd European Conference on Smart Structures and Materials*, 2361:47–50, 1994.

[62] Insurance Institute for Highway Safety. Small overlap frontal crashworthiness evaluation crash test protocol (Version II), 2012.

[63] Insurance Institute for Highway Safety. Status report Vol. 47, No. 6. Technical report, Arlington, VA, USA, 2012.

[64] International Traffic Safety Data and Analysis Group. Road Safety Annual Report 2011, 2012.

[65] A. Jäschke and U. Dajek. Roof-frame design using hybrid technology. *VDI-Report*, 4261:24–45, 2004.

[66] G.R. Johnson and W.H. Cook. A constitutive model and data for metals subjected to large strains, high strain rates and high temperatures. In *Seventh International Symposium on Ballistics*, The Hague, Netherlands, 1983.

[67] M.M. Kamal and J.A. Wolf. *Modern Automotive Structural Analysis*. Van Nostrand-Reinhold, New York, 1982.

[68] B.S. Kang, W.S. Choi, and G.J. Park. Structural optimization under equivalent static loads transformed from dynamic loads based on displacement. *Computer and Structures*, 79:145–154, 2001.

[69] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks, No. IV*, Piscataway, NJ, USA, 1995.

[70] H. Kim. New extruded multi-cell aluminum profile for maximum crash energy absorption and weight efficiency. *Thin-Walled Structures*, 40:311–327, 2002.

[71] S. Kirkpatrick, C. Gelatt, and M. Vecchik. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[72] E. Kita and T. Toyoda. Structural design using cellular automata. *Structural and Multidisciplinary Optimization*, 19:64–73, 2000.

[73] D.G. Krige. A statistical approach to some mine valuations and allied problems at the witwatersrand. Master's thesis, University of Witwatersrand, 1951.

[74] J.P. Leiva. Methods for generating perturbation vectors for topography optimization of structures. In *5th World Congress of Structural and Multidisciplinary Optimization*, Venice, Italy, 2003.

[75] J.P. Leiva. Topometry optimization: A new capability to perform element by element sizing optimization of structures. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, USA, 2004.

[76] J.P. Leiva. Structural optimization methods and techniques to design efficient car bodies. In *International Automotive Body Congress*, Troy, MI, USA, 2011.

[77] J.P. Leiva. A topology optimization method to extract optimal beam-like, plate-like, or shell-like structures from a solid finite element mesh. In *3rd International Conference on Engineering Optimization*, Rio de Janeiro, Brazil, 2012.

[78] Livermore Software Technology Corporation, Livermore, CA, USA. *LS-DYNA - Theory manual*, 2006.

[79] Livermore Software Technology Corporation, Livermore, CA, USA. *LS-TaSC - Topology and Shape Computations for LS-DYNA, v2.0*, 2011.

[80] Livermore Software Technology Corporation, Livermore, CA, USA. *LS-DYNA Version 971 Release 6.1.0, Keyword User's Manual, Volume I*, 2012.

[81] Livermore Software Technology Corporation, Livermore, CA, USA. *LS-DYNA Version 971 Release 6.1.0, Keyword User's Manual, Volume II Material Models*, 2012.

[82] Livermore Software Technology Corporation, Livermore, CA, USA. *LS-OPT User's Manual, Version 4.2*, 2012.

[83] D. Lönn, O. Fyllingen, and L. Nilsson. An approach to robust optimization of impact problems using random samples and meta-modelling. *International Journal of Impact Engineering*, 37:723–734, 2010.

[84] C.L. Magee. Design for crash energy management, present and future developments. In *Seventh International Conference on Vehicle structural mechanics*, Detroit, MI, USA, 1988.

[85] H.F. Mahmood and B.B Fileta. *Vehicle Crashworthiness and Occupant Protection*, chapter Design of vehicle structures for crash energy management, pages 11–110. American Iron and Steel Institute, Southfield, MI, USA, 2004.

[86] J. Marsolek and H.G. Reimerdes. Energy absorption of metallic cylindrical shells with induced non-axisymmetric folding patterns. *International Journal of Impact Engineering*, 30:1209–1223, 2004.

[87] C. Mozumder. *Topometry optimization of sheet metal structures for crashworthiness design using hybrid cellular automata.* PhD thesis, University of Notre Dame, USA, 2010.

[88] C. Mozumder, A. Tovar, and J.E. Renaud. Topology design of plastically deformable structures with a controlled energy absorption for prescribed force and displacement response. In *8th World Congress on Structural and Multidisciplinary Optimization*, Lisbon, Portugal, 2009.

[89] R.H. Myers, D.C. Montgomery, and C.M. Anderson-Cook. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments.* Wiley, 2009.

[90] National Highway Traffic Safety Administration (NHTSA). FMVSS No 208: Occupant crash Protection, 2012.

[91] J. von Neumann. *Theory of self-reproducing automata.* University of Illinois Press, Urbana, IL, USA, 1966.

[92] P.E. Nikravesh, I.S. Chung, and R.L. Benedict. Plastic hinge approach to vehicle crash simulation. *Computers and Structures*, 16(1-4):395–400, 1983.

[93] R. Oftadeh, M. Mahjoob, and M. Shariatpanahi. A novel meta-heuristic optimization algorithm inspired by group hunting of animals: Hunting search. *Computers and Mathematics with Applications*, 60(7):2087–2098, 2010.

[94] C. Ortmann and A. Schumacher. Graph and heuristic based topology optimization of crash loaded structures. *Structural and Multidisciplinary Optimization*, 47(6):839–854, 2013.

[95] G.J. Park. Equivalent static loads method for non linear static response structural optimization. In *9th LS-DYNA German User's Forum*, Bamberg, Germany, 2010.

[96] G.J. Park. Technical overview of the equivalent static loads method for nonlinear static response structural optimization. *Structural and Multidisciplinary Optimization*, 43(3):319–337, 2011.

[97] N. Patel. *Crashworthiness design using topology optimization.* PhD thesis, University of Notre Dame, USA, 2007.

[98] C.B.W. Pedersen. *On Topology Design of Frame Structures for Crashworthiness.* PhD thesis, Technical University of Denmark, 2002.

[99] C.B.W. Pedersen. Crashworthiness design of transient frame structures using topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 193:653–678, 2004.

[100] C.L. Penninger, L.T. Watson, A. Tovar, and J.E. Renaud. Convergence analysis of hybrid cellular automata for topology optimization. *Structural and Multidisciplinary Optimization*, 40:271–282, 2010.

[101] E. Perry, R. Balling, and M. Landon. A new morphing method for shape optimization. In *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St Louis, MO, USA, 1998.

[102] O.M. Querin, V. Young, G.P. Steven, and Y.M. Xie. Computational efficiency and validation of bi-directional evolutionary structural optimisation. *Computer Methods in Applied Mechanics and Engineering*, 189:559–573, 2000.

[103] M. Rais-Rohani, K.N. Solanki, E. Acar, and C.D. Eamon. Shape and sizing optimisation of automotive structures with deterministic and probabilistic design constraints. *International Journal of Vehicle Design*, 54(4):309–338, 2010.

[104] M. Redhe, M. Giger, and L. Nilsson. An investigation of structural optimization in structural design using a stochastic approach. *Structural and Multidisciplinary Optimization*, 127(6):446–459, 2004.

[105] R. Le Riche and R. Haftka. On global optimization articles in SMO. *Structural and Multidisciplinary Optimization*, 46(5):627–629, 2012.

[106] A. Schumacher. Parameter-based topology optimization for crashworthiness structures. In *6th World Congresses of Structural and Mutlidisciplinary Optimisation*, Rio de Janeiro, Brazil, 2005.

[107] A. Schumacher, C. Olschinka, and B. Hoffmann. Topology optimization based on graph theory of crash loaded flight passenger seats. In *LS-DYNA Anwenderforum*, Bamberg, Germany, 2008.

[108] The Scilab Consortium. Scilab. `www.scilab.org`.

[109] SFE GmbH, Berlin, Germany. *SFE CONCEPT Ver 4.2.2.3, Reference Manual*, 2009.

[110] M.K. Shin, K.J. Park, and G.J. Park. Optimization of structures with nonlinear behavior using equivalent loads. *Computer Methods in Applied Mechanics and Engineering*, 196:1154–1167, 2007.

[111] O. Sigmund. On the usefulness of non-gradient approaches in topology optimization. *Structural and Multidisciplinary Optimization*, 43(5):589–596, 2011.

[112] C. Soto and A. Díaz. Basic models for topology design optimization in crashworthiness problems. In *Proceedings of DETC 99*, Las Vegas, Nevada, USA, 1999.

[113] K.C.B. Steer, A. Wirth, and S.K. Halgamuge. *Nature-inspired algorithms for optimisation*, chapter The Rationale Behind Seeking Inspiration from Nature, pages 51–76. Springer, Berlin, Germany, 2008.

[114] K. Takada and W. Abramowicz. Macro element fast crash analysis of 3d space frame. In *SAE World Congress 2007*, Detroit, USA, 2007.

[115] TASS BV, Rijswijk, Netherlands. *Madymo v7.4.1, Reference Manual*, 2012.

[116] B. Tatting and Z. Gurdal. Cellular automata for design of two-dimensional continuum structures. In *Proceedings of Eighth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, USA, 2000.

[117] V. Tereshko and A. Loengarov. Collective decision-making in honeybee foraging dynamics. *Computing and Information Systems Journal*, 9(3), 2005.

[118] A. Tovar, N. Patel, G. Niebur, M. Sen, and J.E. Renaud. Topology optimization using a hybrid cellular automaton method with local control rules. *ASME Journal of Mechanical Design*, 128(6):1205–1216, 2006.

[119] S. Ulam. Random processes and transformations. 2:85–87, 1952.

[120] Vanderplaats Research & Development, Inc, Colorado Springs, CO, USA. *GENESIS v12.2, Quick Reference Manual*, 2012.

[121] R. Vescovini and C. Bisagni. Buckling analysis and optimization of stiffened composite flat and curved panels. *AIAA Journal*, 50:904–915, 2012.

[122] K. Volz. *Physikalisch begründete Ersatzmodelle für die Crashoptimierung von Karosseriestrukturen in frühen Projektphasen*. PhD thesis, Technische Universität München, Germany, 2011.

[123] T. Weise, M. Zapf, R. Chiong, and A.J. Nebro. *Nature-inspired algorithms for optimisation*, chapter Why is Optimization difficult ?, pages 1–50. Springer, Berlin, Germany, 2008.

[124] Wikimedia Commons User: Fropuff. The vertex configuration of a tetrahedral-octahedral honeycomb. `commons.wikimedia.org/wiki/File:TetraOctaHoneycomb-VertexConfig.svg`, 2006.

[125] World Health Organization. Global status report on road safety. Technical report, Geneva, Switzerland, 2013.

[126] P. Wriggers. *Computational Contact Mechanics*. Springer, 2006.

[127] S.I. Yi, H.A. Lee, and G.J. Park. Optimization of a structure with contact conditions using equivalent loads. *Journal of Mechanical Science and Technology*, 25(3):773–782, 2010.

[128] T. Zeguer and S. Bates. Signpost the future: Simultaneous robust and design optimization of a knee bolster. In *Altair Engineering CAE Technology Conference*, Gaydon, UK, 2007.

[129] X. Zhang, G. Cheng, and H. Zhang. Theoretical prediction and numerical simulation of multi-cell square thin-walled structures. *Thin-Walled Structures*, 44:1185–1191, 2006.

Appendix A- Implementation files

A.1 Optimisation input file `hcatws_user_call.sce`

The following file defines all the user parameters necessary to run an optimisation with the HCATWS method. Once the parameters are chosen, the optimisation is started by running this file in the software Scilab. Note that the lines starting with // are comment lines.

```
//---------------------//
// hcatws_user_call_1.5.sce
// hcatws algorithm user parameters and call
// April 2013 - Hunkeler
// calling hcatwsbuild_u_1.5.sce
//---------------------//
//
//
//////// initialisation block ////////
// restart iteration
restart_point=0;
// specific starting point (0=no, 1=yes)
spstartpt=0;
////////////////////////////////////////////////////////////////
// optimisation user parameters ////////////////////////////////
////////////////////////////////////////////////////////////////
//
// starting mesh file
strtmesh="start.key";
// starting SFE CONCEPT model file
strtmodel="cells.SFECmod";
// mass fraction objective
mf0=0.45;
// minimum mass fraction
Mf_min=0.05;
// neighbourhood type
```

```
// (0=empty, 1=plane VN, 2=plane Moore, 3=3D VN, 4=3D Moore,
//  5=extru pbm, 6=extru xhca, 7=extru xhca coupled)
neightyp=7;
// spot weld connections (0=no, 1=yes)
swc=0;
// max spot resultant force (N)
// irrelevant for this case
max_spot_cst=20000;
// update rule type
// (1=Tajs-Zielinska, 2=Patel, 3=Hybrid,
// 4=advanced including spot weld, 5=Hybrid xhca,
// 6=Hybrid 2D xhca full, 7=Hybrid 2D xhca full no reintro,
// 8=HVNwithR, 9=Hybrid 2D xhca full noreintroadv)
upd_typ=9;
// thickness boundaries (mm)
t_min=0.4;
t_max=3.0;
// maximum thickness variation (mm)
max_t_var=0.5;
// mass fraction increment factor and period
// (Mf_inc_fact*exp(-iteration/Mf_inc_tau)*Mf(iteration))
Mf_inc_fact=0.3;
Mf_inc_tau=50;
// number of boxes in each direction
xboxes=11;
yboxes=3;
zboxes=1;
// initial kinetic energy (J)
KE_0=1400;
// minimum setpoint value
S_star_min=0.0001;
// number of cells
nbc=118;
// maximum number of iterations
maxit=200;
// maximum displacement (or force)
maxdisp=45;
// Force objective (0=no, 1=yes)
force_objective=0;
// mass constraint convergence criterion
```

```
eps_1=0.001;
// algorithm convergence criterion
eps_2=0.1;
// symmetry keys
sym_key_yz=0;
sym_key_xz=0;
sym_key_xy=0;
sym_key_0=0;
sym_key_0s=0;
// extrusion length
extru_length=90;
// cells material density
density=2.7e-9;
// base cells width
w_base=30;
// initial model extra mass
mass_init=0.028248;
// paths
routines_path="~/routines/";
concept_path="~/sfe_concept_dir/bin/concept";
compute_and_extract_path="../compute_and_extract.sce";
monitoring_path="../monitoring.sce";
//
////////////////////////////////////////////////////////////////
// end of optimisation user parameters ////////////////////////
////////////////////////////////////////////////////////////////
//
exec(routines_path+"hcatwsbuild_u_1.5.sce");
//
```

A.2 Computation parameters and results to extract `compute_and_extract.sce`

The following file defines the computation parameters (pre-processing, finite element solver call, input file, post-processing...) and the results which need to be extracted after the computation. It is the outer part of the loop detailed in Figure 4.12 and it is called at every iteration of the optimisation process. Depending on the optimisation problem and on the working environment, these parameters can be completely different. The file presented here should only be seen as an example.

```
//------------------------//
// compute_and_extract.sce
// launches ls-dyna computation
// checks for errors
// creates simplified outputs
// May 2012 - Hunkeler
//------------------------//
//
// parameters and path ////////////////////////////////////////
//
// path finite element model
path_main_deck_file="../compiled.key";
// number of processors
n_cpu="4";
// solver executable
path_lsd="/usr/bin/lsdyna/ls971_s_R6_0_0_amd64_suse11";
lsd_exe=path_lsd+" I="+path_main_deck_file+" ncpu="+n_cpu;
////////////////////////////////////////////////////////////////
//
// pre-processing calls ///////////////////////////////////////
//
// in restart mode, only the results are read
if iteration>restart_point then
// perl script to update SFE CONCEPT batch script
unix_s("perl ../updbatchSFE.pl")
// creation of model via SFE CONCEPT
unix_s(concept_path+" -a -b batch_script_SFE.con")
// perl script to update fe-definitions (for diagonal cells)
unix_s("perl ../upd_addings.pl")
//
////////////////////////////////////////////////////////////////
//
// FE simulation and extraction ///////////////////////////////
//
stopcompu=0;
n_compu=0;
n_compu_s=string(n_compu);
// start loop
while stopcompu==0 & n_compu<4 do
unix_s(lsd_exe);
```

```
// check for last output
checkbinfile=isfile("d3plot02");
// run results extraction script
if checkbinfile then
[out,outcheck]=unix_g("perl ../outputfiles.pl");
if outcheck==0 then
stopcompu=1;
end;
end;
n_compu=n_compu+1;
n_compu_s=string(n_compu);
end;
end;
//
////////////////////////////////////////////////////////////
//
// extraction and history (by algorithm) ////////////////////////
//
// extraction of global results by algorithm
GLO_OUT=fscanfMat("global_output.out");
// constraint definition (displacement/force)
Gdisp(iteration)=GLO_OUT(1);
// mass definition
tp_mass=GLO_OUT(2)-mass_init;
////////////////////////////////////////////////////////////
```

A.3 History outputs `monitoring.sce`

This file defines the specific outputs which need to be monitored during the optimisation process. They will be written to the `history` file at each iteration.

```
//------------------------//
// monitoring.sce
// defines output values to be written in history file
// October 2012 - Hunkeler
//------------------------//
mfprintf(hist_file, "real mass=%f \n", tp_mass);
mfprintf(hist_file, "displacement=%f \n", GLO_OUT(1));
mfprintf(hist_file, "front displacement=%f \n", GLO_OUT(5));
mfprintf(hist_file, "new Mf objective=%f \n", Mf(iteration+1));
////////////////////////////
```

A.4 Lists of cells `cells.list` and `delcell.del`

The cell lists are generated by the algorithm at each iteration for update of the model by the CAE software SFE CONCEPT. `cells.list` contains the current thickness of each cell in the model while `delcell.del` is a list of cells which should not appear in the current finite element model.

`cells.list`

```
1010100 1.25
1010102 0.85
1020100 0.35
1020102 0.21

...

8030102 0.39
8030109 0.90
```

`delcell.del`

```
1020100
1020102

...

8030102
```

A.5 Computation output files `global_output.out` and `cells_output.out`

The following files are generated after each finite element simulation. They are specially formatted to gather the useful information for the algorithm. `global_output.out` contains the global outputs of the model, they are used to update the mass fraction objective. `cells_output.out` contains the outputs specific to each cell of the model.

`global_output.out`

```
41.45            -> displacement output (mm)
0.250E-01        -> mass of the model (tons)
```

```
5.000E-06        -> added mass with mass scaling (tons)
0.002            -> added mass percentage
9.22073E+02      -> resultant kinetic energy (mJ)
1.66739E+06      -> resultant internal energy (mJ)
4.95016E+03      -> resultant hourglass energy (mJ)
2.50962E+03      -> resultant sliding interface energy (mJ)
1.67075E+06      -> total energy (mJ)
1.00701E+00      -> total energy over initial energy
```

cells_output.out

```
cell/Internal Energy (mJ)/added mass (t)/mass (t)
1030109    21.556           0.00E+00    0.773E-05
2010108    0.1549           0.00E+00    0.412E-05

...

8010100    6.0611           0.00E+00    0.408E-05
```

Appendix B - Contact definitions for diagonal enriched 2D space filling

As an illustration to Section 6.1.2, an example of the different contact definitions related to the diagonal enriched 2D space filling is given here. The model considered here is made of the following list of non-diagonal cells: $\forall\, 1 \leq i \leq 9$, $10i$, $10i+2$, along with the following list of diagonal cells: $\forall\, 1 \leq i \leq 9$, $10i+8$, $10i+9$. In this case, the diagonal cell $10i+8$ intersects diagonal cell $10i+9$. As explained in Section 6.1.2, the contact definitions are treated differently for non-diagonal and diagonal cells.

Contact $\boxed{1}$ is the single surface contact definition between the active non-diagonal cells amongst: $\forall\, 1 \leq i \leq 9$, $10i$, $10i+2$.

If cell $10i+8$ is active, contact $\boxed{10i+8}$ is defined as a surface to surface contact between cell $10i+8$ and all the active cells amongst: $\forall\, 1 \leq j \leq 9$, $10j$, $10j+2$ and the active cells amongst $\forall\, 1 \leq j < i$, $10j+8$, $10i+9$.

Similarly, if cell $10i+9$ is active, contact $\boxed{10i+9}$ is defined as a surface to surface contact between cell $10i+9$ and all the active cells amongst: $\forall\, 1 \leq j \leq 9$, $10j$, $10j+2$ and the active cells amongst $\forall\, 1 \leq j < i$, $10j+8$, $10j+9$.